Eastern Economy Edition

BIG DATA ANALYTICS

Edited by
Parag Kulkarni • Sarang Joshi
Meta S. Brown

PHI

Big Data Analytics

Big Data Analytics

Edited by

PARAG KULKARNI
Founder and CEO
iknowlation Research Labs

SARANG JOSHI
Professor
Pune Institute of Computer Technology (PICT), Pune

META S. BROWN
President
A4A Brown Inc.

PHI Learning Private Limited
Delhi-110092
2016

₹ 250.00

BIG DATA ANALYTICS
Edited by Parag Kulkarni, Sarang Joshi and Meta S. Brown

ISBN-978-81-203-5116-5

The export rights of this book are vested solely with the publisher.

Published by Asoke K. Ghosh, PHI Learning Private Limited, Rimjhim House, 111, Patparganj Industrial Estate, Delhi-110092 and Printed by Mudrak, 30-A, Patparganj, Delhi-110091.

Contents

Preface

Scientist, engineers and researchers rarely come together to write something of use. Anyway it is not easy when ten different researchers come together and write on same topic. We are sure that it takes lot of understanding to make sense out of it. Probably, this is not applicable to this book, and we have tried to simplify some aspects of Data Mining and Big Data together. There are numerous books on Big Data, and we are sure that they offer great bit of insight into this very important topic. Then obviously question arises why there is another book. The question is relevant, and here we have answer to this question. This book emerged as an outcome of research by different researchers working in area of Data Mining. Dr. Parag Kulkarni and PhD candidates working/worked under his supervision joined hands with Meta S. Brown (Author of *Data Mining for Dummies*) and Dr. Sarang Joshi to produce this different sort of book. While working on relevant topic, each researcher has explored, researched and practiced specific aspects of Data Mining and Machine Learning. In this book, he/she tries to put it in big data perspective. What is big data? At the end of the day, it is about size. In the world where size matters, big data became really a big and valuable term. People built prolific careers out of it, companies built fortunes out of it, probably visionary countries will build astounding economies out of it. Sometimes the terminologies and descriptions become so verbose that those frighten researchers and sometimes so repetitive that they loss their meaning. This book is an attempt to give meaning to these words with simple unstructured data mining in this clamorous word war. This book tries to establish relevance for these terms. Big Data—nothing big about it. It is more about handling large sized unstructured data and elegance to deal with variety of data arriving at great pace. This book is intended to readers who are looking for a big data trends and its relationship with traditional data mining. The theme is big data and mining unstructured data. The book tries to showcase the result of research by the team in the last five years, where we worked extensively on unstructured data mining and its extension to big data mining. Every chapter of this book presents new aspects of unstructured data mining and text analytics. This book elaborates relationship between text analytics and big data with reference to practical problems and research carried out in this area. Chapter 1 takes overview of the topic and touches some recent trends.

Chapter 2 introduces various data mining methods and models along with different applications. This chapter gives a platform to proceed to different big data mining related concepts. This chapter also discusses practical aspects with case studies.

Chapter 3 deals with big data and different methodologies. This chapter, with detailed examples, discusses mining of big data using different tools. Chapter 4 is about context, which

is a very important piece of information, which is known partially. Application of context in unstructured big data is effective. This chapter covers how to use context enabled data, the challenges in using context and how to find context in long and short text.

Chapter 5 discusses the concept of big data text categorization and topic modelling. It introduces the concept of context-based learning by exploiting context at hyperlink and linguistic level. It also highlights the relation extraction and the usage of GATE tool. Further, it introduces the techniques of topic modelling. Later situation model is discussed for building situations from text using Wordnet and similar measures.

Chapter 6 discusses multi-label text categorization from big data perspective. In text analytics, a single text document may belong to multiple concept classes simultaneously because of inherent ambiguity existing in text representation. Inferring knowledge from such a scenario is known as multi-label. This process makes overall classification and association process more complex. Moreover, in big unstructured data, this multi-label text categorization problem becomes even more difficult to solve. Due to simplicity and large applicability, graph representation found to be most suitable representation for text document. These graph representations retain information such as ordering and association of terms. Different graph algorithms are useful for text analytics. Since this is one of the most relevant problems in text analytics with reference to big data, Chapter 5 is indebted to address the aforesaid challenges by discussing various issues in multi-label unstructured big data mining.

Distributed clustering has become extremely important pre-processing task in mining distributed data sources. Many of the real world distributed datasets consist of objects modelled by high dimensional data, e.g., image retrieval, molecular biology, information retrieval and so on. Thus, in Chapter 7, we introduced various challenges involved in distributed as well as high dimensional data clustering. Subspace clustering algorithms look for and build overlapping clusters, not necessarily in the whole dimension space, but also in subspaces of the attributes. Since this is the best solution available, to find clusters hidden in high dimensional distributed data, Chapter 7 further details the subspace clustering methodology best suitable for big data.

Chapter 8 covers the basic concepts of machine learning and different learning paradigms. The necessity and importance of ML techniques in the analytics for prediction and forecasting are discussed. The chapter covers need of incremental approach in the analysis of Big Data with applications to the same. Chapter 9 includes the aspects of data analytic to create value. It also covers some important aspects of business analytics. Chapter 10 concludes the discussion. It summarizes few important aspects covered in the book giving few pointers for more thinking. Annexure I gives introduction to Hadoop framework from big data perspective.

We are sure that this book will prove as an important and very useful addition to the literature in this space.

Parag Kulkarni
Sarang Joshi
Meta S. Brown

Acknowledgements

This book is a research outcome resulting due to collaboration among researchers. We are thankful to all contributors of this book. We also take opportunity for extending our gratitude to Savitribai Phule Pune University, Devi Ahilya Vishwa Vidyalaya and different researchers from these universities who directly or indirectly contributed to this journey. We are thankful to our family members, friends and reviewers. We also take opportunity to thank great researchers who created new research avenues in areas of Data Mining, AI, Machine Learning and Big Data. We are thankful to our PhD candidates, and ME/MTech students who directly or indirectly contributed to this effort. Big Data has opened new doors of economics.

We are also thankful to the editorial team of PHI Learning, specially Mr. Malaya R. Parida (Acquisition Manager) and Lakshmi, for their perseverance and patience to deal with this assorted text to convert it into a wonderful book. We thank Alesia Siuchykava for connecting us. We take opportunity for extending our gratitude to COEP, PICT, GH Raisoni College, Cummins College and D.Y. Patil College for their support during this endeavour. We are thankful to Dr. P.T. Kulkarni, Dr. (Mrs.) M.B. Khambete, Dr. R.K. Jain, Dr. R.D. Kharadkar and Dr. Anil Sastrabudhe for their valuable support. It is an interesting journey, and we take opportunity to thank all faculty members and students who contributed to this journey.

Parag Kulkarni
Sarang Joshi
Meta S. Brown

1

Introduction to Big Data

—DR. PARAG KULKARNI

◊ 1.1 INTRODUCTION

14/10/2018

The world is a complex system. Everyday numerous transactions and events take place in everyone's life. These events contribute to data building and collection. This data generated during every transaction are not in specific format—is not structured and is coming from various sources. These events are related to other events—and hence lead to chains of events. This brings huge data in front of us—rather huge semi-structured or unstructured data. Though some structured data is also part of whole chunk—the percentage of structured data is negligible. Making best use of this data for decision-making is the key. Collecting, analyzing and using this data are the major challenges in front of us. Text analytics, business analytics, and software analytics rather data analytics is about analyzing trends in this data and building insights. This huge unstructured data mining and processing is the theme of this book. The learning and mining methodologies for small datasets focussed on structure of data are simply not capable of coping up with this Big Data problem. The conventional learning techniques are based on too many assumptions. These assumptions are not true in real life while dealing with huge data.

The basic assumption here is—this unstructured data processing and text analytics will solve the problems faced due to traditional data mining and processing. But no solution comes without challenges. Big Data analytics and Big Data mining pose many challenges due to size of data, speed of processing required and heterogeneity of data. This chapter takes overview of this journey of mining unstructured data.

◊ 1.2 WHAT IS BIG DATA?

In last couple of years, everyone is talking in big way about Big Data. World began running after Big Data. There are many applications where we need to process large amount of data. This data comes in many forms but mainly in unstructured form. Right from crowd behaviour, big communities, and social networking sites, there are many real life scenarios where huge

amount of data is generated and that needs to be mined. Human being and all other living things and even non-living things on this earth have been generating data for thousands of years. This data include behavioural data, transactional data, associative data and what not. All enterprises and societies are generating data—in different forms, and data is an integral part of all these enterprises and societies. This data comes in different forms. Enterprises capture data about customers, sales, products, financial transactions, profit and so on. Collecting this data, processing this data, storing and mining this data and finally using this data effectively for decision-making are the objectives of data mining and data analytics research. Theoretically, using this data to build competitive advantage and to provide better services looks perfectly fine. But handling and using this huge unstructured data include many challenges. These challenges range from handling of data types to data size. The data has very high volumes, different types and varieties and further it is coming with very high pace. On top of that in real life problems we expect this data to be handled and processed with very high efficiency, accuracy and pace. To deal with this data and to manage and leverage this data of huge size, high pace and large varieties of different technologies come together and converge in the form of Big Data. Big Data is one that deals with this problem and allows users to gather, store, manage, manipulate and mine this huge data.

Mining of Big Data needs going beyond traditional unstructured data mining—it is about association and deriving broader patterns. In short, Big Data is not a single technology used for data mining, rather it is a combination of all technologies those come under umbrella of unstructured and structured data mining. Big Data mining is about handling variety of data, higher velocity of data and large volume of data to timely derived data—data patterns available for decision-making.

This book provides different perspective on Big Data and unstructured data mining. In last couple of years, many books are written to unfold Big Data mining concepts. This book is an unstructured data-mining safari, which will take us through different aspects of unstructured data mining while unfolding different practical aspects of Big Data mining. This book will also focus on Machine Learning (ML) and mining methods required for processing and decision-making in case of Big Data.

1.3 MINING UNSTRUCTURED DATA: Challenges and Modern Techniques

Most of the data available in the world is unstructured. While there are a few challenges while managing structured data, the challenges increase to multifold when it is unstructured. Mining unstructured data is challenging due to following reasons:

1. There are no labels of any sort
2. It is very difficult to clean the data
3. Deriving a model and picking useful data are difficult task.

Business needs behaviours of their customers, stakeholders and even other entities associated with it. There are a number of information exchanges among number of customers, shops, and there are myriad transactions carried out in whole process. There are millions and

millions email exchanges among organizations and customers, there are postings on different blogs, there are exchanges and posting on social networking websites, bulletin boards, tweets and so on. All these data are in unstructured form, and it is almost impossible to manually compile and mine this data to draw conclusions. Another possible drawback of manual method is time required to process the data. Thus, providing solutions in time with manual methods is almost impossible. These data possess unique capability to give rich insight into customer behaviours those are very important from business perspective.

There can be many such problems those required processing of unstructured data. Due to lack of any structure, heterogeneity and possibility of impact of surrounding information and context, it becomes very complex task to analyze this unstructured data and arrive at conclusion. Unstructured data mining is highly knowledge-intensive process. There are multitude reasons for complexity of unstructured data mining like seeking useful information in case of unstructured data mining is multi-dimensional task and needs to consider users' interest. Because of many such possibilities, it is challenging to explore interesting patterns and association among them. There are though many similarities between structured and unstructured data mining like requirement of preprocessing and pattern discovery, unstructured data mining demands different methods and explorative intelligence to handle uncertainty, dynamic behaviour and inference.

Some researchers think that unstructured is a misleading terminology. Data is either semi-structured or weakly-structured. All types of text documents have some sort of semantic structure and that is even true for other types of documents. Documents or data that have very little strong typographical layout, or markup indicators represent structure, like most research papers, legal and government documents, stories and even randomly collected data, are examples of weakly structured documents or data collections.

There are many concepts beyond tokens, words and characters, those try to explain similarity and theme of documents. As we go through this book, we will elaborate these ideas. There are documents and data features like concepts, context, theme, topic, and topic representation.

Concepts: Concepts are properties of documents those are evident through typical statistical and rule-based categorization. The concepts may not be directly about occurrence of particular keyword or key phrase. It is more about the concept the document is trying to represent. A document may represent healthcare concept without mention of healthcare even once. Similarly, document may represent concept of nutrition without mention of actual word. Concept goes beyond the actual occurrence of word. Concept identifier tries to find out the concept based on occurrence and association between tokens.

For instance, a document collection that includes reviews of sports cars may not actually include the specific word 'automotive' or the specific phrase 'test drives,' but the concepts 'automotive' and 'test drives' might nevertheless be found among the set of concepts used to identify and represent the collection. Concept can help in clarifying and disambiguating occurrence of words. Unlike traditional word- and term-level features based on occurrence and frequency of terms, concept-based features can consist of phrases, words or even corpus of words, not specifically found or occurred multiple times in document.

Context: It goes beyond document and looks for the context of that document, context of event, and its importance with reference to user context. Context can be about place, time,

theme and situation. Something may be important in a particular context, but may not be relevant in some other context.

Theme: The idea that represents the text—or the idea which is in alignment with text. Theme can bring various documents together in cluster. Thematic classification can help in getting cluster of document for decision-making. Theme is more intrinsic property while topic is more of representative property, and hence themes can be used to catch subtle difference with reference to domain and application. In some cases, theme is referred as what important words are used in the representation.

Topic: Topic is rather prominent theme or a single representative idea of the text. In some cases theme and topic both words are used for subject of discussion of composition. But from classification and decision-making perspective, topic is more generic while theme is more specific.

◇ 1.4 UNSTRUCTURED DATA MINING APPLICATIONS

Since most of the data and data available in different domain available in unstructured form, most of the applications demand unstructured data mining. From different data streams coming as output form different analysis, output produced by different machines, documents produced for different applications like legal, health care, banking and insurance, there are lot of unstructured or semi-structured data. There are numerous applications of unstructured data mining including:

- Analysis of legal documents by lawyer
- Analysis of patent documents by patent attorneys
- Analysis of patients data and behaviours
- Opinion mining
- Business data analysis

Unstructured data mining includes the following:

1. Search and data extraction
2. Document analysis, collation and management
3. Business intelligence
4. Opinion mining

Unstructured data mining is not a single discipline and requires various scientific disciplines to work together like:

1. Machine learning
2. Statistics
3. Natural language processing
4. Text processing and mining
5. Linguistic and association

1.5 BIG DATA ANALYTICS: Challenges

Big Data is heterogeneous and contains different flavours. Hence, Big Data is bit confusing to handle and its analysis is very complex. Big Data analytics finds out patterns, association among variety of data with business outcomes. The challenges associated with Big Data mining and Big Data analytics are different from other data, since it requires higher pace and more efficient algorithms.

Challenges in Big Data analytics include—It is heterogeneous—The natural languages are feature rich. Big Data is heterogeneous, but typical traditional processing algorithms expect homogeneous data. Further, there is practical difficulty that all data cannot be available. For example, if an employee fails to provide all data, some of the fields are missing. Dealing with this partial information or incomplete information is another challenge in mining and processing of Big Data. Even use of error handling methods could not handle some of such cases.

The increasing volume of data is another challenge while analyzing Big Data. Though using cloud computing we can store large amount of data, there is a demand for timely response with reference to interactive and distributed processing of data.

1.6 ADVANCED MACHINE LEARNING AND TEXT DATA MINING

With so many possibilities and need of intelligent data handling for unstructured data, it demands learning methods with different capabilities to analyze unstructured data. To accommodate different conceptual, contextual and thematic features and to give most appropriate decisions, traditional ML is not sufficient. There is a need to find association among different themes, contextual fusion, concept modelling, representation and processing of context vectors.

Since unstructured data keeps coming and it is huge in size, traditional learning technique based on structured historical patterns does not serve the purpose. There is need of learning in different way to handle dynamic behaviour, size and hidden relationships in case of unstructured data. Typically, incremental machine learning, adaptive machine learning, and advanced clustering techniques based on exploration are required to handle unstructured data. In this book, we will try to elaborate these methods and types with reference to standard datasets as well as custom data set.

1.7 WHAT IS CONTEXT?

Context refers to environmental and situational importance and positioning of data, object or document. Context decides importance, even meaning and relevance of action or data. Something that is important at one place may not be important at other place, something that is relevant today may not be relevant tomorrow, and something that is very important for a particular person may not be that important for other person. Context tries to capture this association. A scientist may look at same data in different context than that of a businessman. Hence, context is not just about document or data, but it is also about its association with user,

environment or a particular domain. Context is about association between text and situation. There can be multiple contexts for a document with reference to environment. There can be local as well as global context of document.

For an example, if the word ball is used with reference to cricket, it may refer to cricket ball, while in case of soccer, it refers to football. That is even relevant in case of numbers, for example, in context of body temperature, 37 is very high while in context of marks it is very low. There are many facets of context. For example, every question has a context. A same question asked in different context will return different answer, for example:

1. Doctor asks patients giving thermometer, 'Tell me what is the temperature'? In this context, temperature refers to body temperature.
2. 'It is too hot today. What is the temperature?' Here temperature refers to room temperature.

Context can be determined based on adjacent sentences; it can be determined based on persons taking part in conversation. It can be based on event took place in recent past.

◇ 1.8 CONTEXT BUILDING THROUGH MULTI-LEVEL DATA MINING

Simple and traditional data mining fail to determine the context. Context is neither a topic and not it is just a class. Context is situation specific, application specific and location specific. Multi-level association rules are used in some cases in literature for larger datasets. Multi-level association helps to reveal different aspects and relationships among datasets. Typically, relationships those are not visible at one level may be visible at other level. There can be association among articles in shop. There can be association among shops. There can be association among localities of different shops. But multi-level association is not just about it, but it is about interdependencies among associations at different levels.

Market basket analysis for unstructured data

There are various methods to analyze data. We will go through these methods with reference to Big Data. Market basket analysis refers to analyzing association among different items in shop based on tendency of customers buying them together. It is a statistical approach. Apriori algorithm, M.S. Apriori algorithm and other improved statistical techniques are used for this analysis. For text analytics, the same method used by researchers is based on co-occurrences of different terms. The extended bag of word technique is used. For larger data size and data coming from different sources where the problem space is increasing, multi-level Apriori algorithm is used. These are quite a few extensions to it. Practically, looking at the Big Data problem, how can one go for association among data points? Can traditional market basket analysis method or modified market basket analysis serve the purpose? Look at this from computational complexity purpose, you will find many issues. How market basket analysis can be improved to meet text analytics and Big Data analytics requirement will also be discussed in subsequent chapters.

◇ 1.9 BUILDING APPLICATION AND DEALING WITH BIG DATA

There are many examples where huge data of heterogeneous nature can be collected. There can be big fair with presence of many human beings like big processions and gatherings, huge data coming from different sources may be data from big cities, data of huge number of transactions like information exchange on mobiles, messengers and social networking sites. Analyzing this huge data—mining it for relevant information for decision-making—is an example of Big Data mining. Dealing with this Big Data is a challenge. Many real life applications are Big Data applications.

Big Data future in healthcare

Social media and social networking have increased connectivity and communication. The different messages are going in different directions and in unstructured or semi-structured formats. Social medias have impacted healthcare industry in great way. This has increased communication among patients, healthcare service providers and communities. The communication between patients and service providers allows the information to flow and is a source of Big Data. Social networking deals with large volumes of data. Large volume of unstructured data with different flavours poses many challenges. There can be different opinions of different patients, there can be biases, misunderstanding and even information displayed based on partial knowledge.

Similarly, there are many challenges in enterprise applications. Actually, all the businesses are turning into information-driven businesses including logistics, healthcare, inventory management and sales analysis. Big Data can enable huge saving in various domains across the globe including healthcare. In healthcare, Big Data can help in healthcare and research domains. Even there are numerous Big Data applications in public sector and governance, government sector needs large data to be processed for effective decision-making and management. Big Data is not just about acquisition of huge data, but it is redefining the landscape of data management, and organizing unstructured data in big-data applications.

◇ 1.10 BIG DATA AND LEARNING

Big Data involves learning. Big Data deals with huge unstructured or semi-structured data. This data is typically heterogeneous, partial, and demands quick results. Hence, Big Data needs better learning methods and needs to handle uncertainty. In fact, applications like document retrieval and medical and healthcare data analysis include data in different formats. For Big Data, due to uncertainty and heterogeneity in data, simple pattern-based methods do not work. Big Data forms domains like atmospheric sciences data: rapidly ballooning observations (e.g., radar, satellites, and sensor networks), continuous electricity data, climate models, ensemble data, etc. We cannot just rely on historical data-based exploitation based methods but we need exploration based methods also. Along with statistical ML techniques like Bayesian networks, Random Forest, probability based techniques working on historical patterns, other methods of association and exploration need to be used.

Some researchers believe that Big Data needs large scale machine learning. This has large number of dimensions, large number of tasks and large outcomes. If we think it in terms of

an intelligent agent, then it has many inputs received through sensors and large processing to produce various actions for actuators. Typical example of this is bioinformatics. This involves computational and statistical challenges. In this direction in recent years many researchers work on evolving methods to handle large data. The work done includes large scale supervised learning, various unsupervised learning and clustering methods for large scale data.

◇ 1.11 ANALYTICS AND BIG DATA

Analyzing Big Data helps to build competitive advantage for organization. Recommendation engines of different e-commerce and search companies in some way or other are working on Big Data analysis. Thus, ability to analyze Big Data provides unique opportunities for organizations in terms of strategic decision-making to build competitive and business advantages. Just sampling large data for analysis may lead to lot of errors but ability to look into real data of huge size can unfold and reveal the real picture.

Basic analytics: Basic analytics refers to traditional methods of data analysis. These methods include dividing data into relevant parts. The data is divided into smaller parts. The smaller set of data is easy to explore and analyze. For example, if we have data across the country, we divide it into smaller chunks to analyze it. Even different dimensions of importance are considered during different phases. This actually makes you unaware of actual problem space. The basic analytics even includes basic monitoring of data in real time. Another approach in basic analytics is anomaly detection. Here, data is observed to detect anomalous events. This uses simple methods based on statistical signature, moving averages or some simple statistical measures. In case of anomaly, the alert is raised.

Advanced analytics: Basic analytics may not be able to handle complex situations. Hence, advanced analytics is used for analyzing unstructured data. The text analytics is used to process unstructured textual data and to bring it to certain form where insight into it can be sought. This uses various methods in computational linguistic, NLP, statistics and other allied branches of computer science. Other analytical and data mining algorithms (hybrid approaches) are also used in this case.

◇ 1.12 TEXT ANALYTICS AND BIG DATA

Large number of unstructured textual information is generated everyday. It is in the form of mails, messages, notifications and documents. This information is heterogeneous and comes from different sources. Most of the time, information is partial and has some sort of bias. This ton and tons of information is at our disposal everyday. There are sentiments about brands, products, news and even about conversations. These sentiments are floating around social media in the form of messages and conversations. Monitoring and looking at these public conversations and opinions on social networks about brands, products, news and events demand higher insights and throughputs from text analytics tools. There are numerous technical aspects of text analytics. This book intents to go into much deeper of text analytics and Big Data to

reveal practical side of text analytics with detailed description of research and application of text analytics and Big Data analytics.

The purpose of text analytics is very clear—it is not mere insight into text rather it understands meaning of words/corpus of words in given context and with reference to context. There can be ambiguity in sentences and in isolation getting the right meaning out of it is difficult. 'He saw a scientist with telescope'. It can have two meanings—either he has telescope or the scientist has telescope. But common sense suggests that most probably the scientist has telescope, still it is not very clear. Many such ambiguities need to be taken care of in text analytics.

Text analytics is intended to build an association among texts in the form of subject framework, which we can visualize in given context. There are number of text analytics research initiatives and analytical functions those come in picture. This chapter elaborates detailed text analytic functions with practical examples and association with bigger picture. This includes:

1. Topic identification
2. Understanding and mining concept
3. Multi-label text categorization and association
4. Multi-document association and summarization
5. Multi-level and distributed clustering
6. Computational linguistic
7. Context determination and association (Context vector machines)
8. Incremental learning and explorative text analytics

The practical example and case studies are provided with intent so that text analytics and unstructured data processing would allow one to understand conversations taking place and data generated in socio spheres. This includes blog posts, tweets, reviews, etc.

Text analytics can be thought of text pre-processing for text mining. It helps in discovering relationships and additional structures in unstructured data. Actually speaking, there is no purpose of converting unstructured data into structured data. Rather text analytics is about using unstructured data and transforming it into usable form. Some of the case studies covered in this book are as follows:

- Sentiment analysis that analyzes opinions about daily news and presents them as per readers choice suppressing negative news
- Monitor brand reputation
- Determine behaviours of customers
- Identify complaints related to products
- Bringing surveys to useful conclusions
- Text analytics to improve customer service
- Book reviews

There can be other business benefits like customer retention, predicting customer behaviours, and improve customer satisfaction.

Topic identification

This work refers to collection of unstructured data and identifying topic and clustering them based on topic. This work also includes association topics and provides bigger picture for business level decision making. Case studies of sentiment analysis and opinion mining are presented in this book.

Concept mining

Lot of work is being carried out on concept mining. The idea is that the concept can be used to understand relationships among documents. Mere conversion of words to concept does not work effectively, hence the book gives detailed work carried out to use contextual information, metadata and association to determine the concept.

Multi-label text categorization

In the world, a single data item can have multiple labels. This makes overall classification and association process more complex. In big unstructured data, this multi-label text categorization problem becomes even more difficult to solve. Since this is one of the most relevant problems in text analytics with reference to Big Data, it is covered in this book.

Multi-document association and summarization

There are large number of documents and unstructured messages. It is necessary to summarize and associate them. This book also covers methods for associating and summarizing these documents.

Multi-level and distributed clustering

Big Data analytics demands the high dimensional data clustering. The data is distributed—you cannot expect this huge data available at a single place. Some techniques like sub-space clustering and its other variants can be suitable for this. Making sub-space clustering more suitable for Big Data analytics, text analytics and research in this direction are elaborated.

Computational linguistic

For sentiment analysis and analysis of business reviews, it is necessary to process huge text. Computational linguistic is a major part of text analytics. It deals with processing of natural languages, presenting and using cognitive capabilities, translation and summarization. This chapter highlights research in this area with reference to text analytics and Big Data. The case studies like opinion mining, mortgage document mining, legal document mining and political opinion mining are discussed. Some key results and applications are also shows as case studies.

Context determination and association (Context vector machines)

Context is the key of modern text analytics and decision-making. Meaning of any sentence is

generally driven by context. Meaning of same sentence in different context can be different. Context determination is one of the most complex tasks. Methods like positional significance, NLP-based methods, word association and term frequency are employed by various researchers. The book also presents a novel approach of context vector machines to determine context and mine text data. This method takes advantage of positional significance and other text association methods while building context vectors. This method helps in processing large sets of unstructured documents.

Incremental learning and explorative text analytics

As size of the data increases, the dimension of data also increases. This increase in dimension increases the computational complexities. Traditional learning methods like learning from scratch or methods not considering historical data at all while learning are no longer effective. This book also introduces various incremental learning methods to accommodate new data during exploration process. It also includes case studies from health care data, textual data, and business data.

◇ 1.13 UNDERSTANDING TEXT ANALYTICS

We can find roots of text analytics in Natural Language Processing (NLP), data mining and knowledge discovery. The techniques of text analysis and extraction are based on computational linguistic. Text analytics is not just about text search. Search typically targets of locating document which user already knows. Text analytics is more about information retrieval and discovering information. NLP provides analysis of text at different levels. This typically includes:

Lexical analysis: It works on characteristics of different individual words.

Syntactic analysis: It uses grammatical structures and syntactical features.

Semantic analysis works on meanings while next level analysis attempts to determine the meaning beyond words and sentences.

◇ 1.14 BUSINESS INTELLIGENCE (BI) PRODUCTS TO HANDLE BIG DATA

The business intelligence products are developed to seek more insight into data and take intelligent decisions for business. The term Business Intelligence (BI) includes the tools, processes and systems those help in the strategic planning process of businesses. It allows a business to collect, store, access and analyze corporate data for presentation in useful form for decision-making.

Business intelligence systems are used in various areas like customer profiling and support, market analysis, statistical analysis, and inventory and distribution analysis, etc. Traditional business intelligence systems are built with reference to small data and predefined inputs, rather more structured and well-understood data. Practically speaking, the traditional

BI systems were not built considering Big Data. On the other hand, Big Data is heterogeneous—is mixture of unstructured and semi-structured data. The high complexity, uncertainty and incompleteness separate it from data that is required for traditional BI systems. This data comes from multitude of sources and consists of lot of noise, variation and missing data points. It can be real time data and hence demands timely response. Hence, traditional tools could not cope up with Big Data. The new BI systems should handle Big Data and hence there is a need of ability of Big Data analysis and mining. While old tools are becoming obsolete in new context, the new BI tools are becoming available, trying to meet these requirements. The modern data discovery tools are being designed to handle Big Data. The data security and privacy solutions also need to be enhanced to cope up with Big Data. A significant part of data collected by business houses is mostly in textual format which includes business communications, text documents, etc. Text analytics deals with document association and representation. Modern BI systems are supposed to deal with this unstructured data.

◇ 1.15 UNSTRUCTURED DATA MINING AND CLASSIFICATION METHODS

Unstructured data mining and learning methods are different than those used for structured data. The structured methods to retrieve values of fields and information do not work for unstructured data. Let us take a simple example of classification of human behaviours at big processions from security perspective. There are millions of people visiting for religious purpose. Hence, safety and security are very important. Behavioural analysis based on sampled data may not serve the purpose, since a single anomalous behaviour without scanned leads to security hazard. Hence, it becomes important to collect all data of behaviours from various inputs. Those may include videos captured across the place, human interactions, social media exchanges, phone calls, and many other sources. This builds huge heterogeneous data. Then we need methods those can classify and associate this data to understand security hazards.

◇ 1.16 BIG DATA AND MACHINE LEARNING TRENDS

Traditional data mining and machine learning are event-based, and structure-based. The scope of data is kept limited to reduce dimensionality and computational complexity. This is at the compromise of systemic information. The holistic picture is not available. In case of big enterprises, huge sales data or even social events like fairs, big gatherings like *kumbh mela*, or business meets, the use of partial data could not give the complete picture resulting in outcomes those could not hold for system. The real life scenarios are data rich, dynamic, uncertain and full of partial and imperfect information. If we look at Big Data from this perspective, it is acquisition and processing huge multi-perspective data to build holistic picture. The simple data and event-based learning and decision-making result in many side effects across the system. Hence, ML and Big Data trends are about addressing this problem. Hence new trends in area of Big Data are not just about collection and data building but along with that analytics, pattern association and decision-making. The ML trends include:

1. **Adaptive ML:** The changing dynamic scenario does not allow the same traditional methods for learning. The learning methods need to adapt new data and new scenario. Adaptive ML is about amending learning methods and strategies with reference to learning scenario and data.
2. **Incremental ML:** There is a need to incremental learning. Each time with advent of new data, we cannot afford to learn from scratch. In light of new scenario, the new method is implemented.
3. **Multi-perspective ML:** Since data is coming from various sources, sometimes data is even incomplete or partial. Even data collected is also collected from different perspective. Decision-maker needs decision from a particular perspective. Multi-perspective ML is about considering different perspectives, analyzing data from different perspectives and providing decision with reference to most appropriate perspective.
4. **Associative ML:** Association is the key in large data sets. Associative ML provides decision by associating different scenarios and data points. It is about associating pattern for pattern analytics. The associative ML is one of the most powerful ways of ML. Here multi-level association among data points and patterns is used for decision-making.
5. **Systemic ML:** What should be the scope of data and what should be the scope of learning environment, due to possibility of increasing complexity? Systemic ML is about learning with reference to system. It focuses more on interdependencies.

◊ 1.17 THIS BOOK

This book is intended to readers who are looking for a Big Data trends and its relationship with traditional data mining. The theme is Big Data and mining unstructured data. The book tries to showcase the result of research by the team in last five years, where we worked extensively on unstructured data mining and its extension to Big Data mining. Every chapter of this book presents new aspects of unstructured data mining and text analytics. This book elaborates relationship between text analytics and Big Data with reference to practical problems and research carried out in this area.

Chapter 2 introduces various data mining methods and models along with different applications. This chapter sets platform to proceed to different Big Data mining related concept. This chapter also discusses practical aspects with case studies.

Chapter 3 introduces Big Data and different methodologies. This chapter with detailed examples discusses mining of Big Data using different tools. Chapter 4 is about context, which is a very important piece of information, which is known partially. Application of context in unstructured Big Data is effective. This chapter covers how to use context-enabled data, the challenges in using context and how to find context in long and short text.

Chapter 5 gives introduction to the concept of Big Data text categorization and topic modelling. It also introduces the concept of context-based learning by exploiting context at hyperlink and linguistic level. It also gives introduction to relation extraction and the usage of

GATE tool. Further, it introduces the techniques of topic modelling. Later situation model is discussed for building situations from text using Wordnet and similar measures.

Chapter 6 discusses multi-label text categorization from Big Data perspective. In text analytics, a single text document may belong to multiple concept classes simultaneously because of inherent ambiguity existing in text representation. Inferring knowledge from such a scenario is known as multi-label. This process makes overall classification and association process more complex. Moreover, in big unstructured data, this multi-label text categorization problem becomes even more difficult to solve. Due to simplicity and large applicability, graph representation found to be most suitable representation for text document. These graph representations retain information such as ordering and association of terms. Different graph algorithms are useful for text analytics. Since this is one of the most relevant problems in text analytics with reference to Big Data, Chapter 6 is indebted to address the aforesaid challenges by discussing various issues in multi-label unstructured Big Data mining.

Distributed Clustering has become extremely important pre-processing task in mining distributed data sources. Many of the real world distributed datasets consist of objects modelled by high dimensional data, e.g., Image Retrieval, Molecular Biology, information retrieval and so on. Thus, in Chapter 7, we introduced various challenges involved in distributed as well as high dimensional data clustering. Subspace Clustering algorithms look for and build overlapping clusters, not necessarily in the whole dimension space, but also in subspaces of the attributes. Since this is the best solution available, to find clusters hidden in high dimensional distributed data, Chapter 7 further details the subspace clustering methodology best suitable for Big Data.

Chapter 8 covers the basic concepts of machine learning and different learning paradigms. The necessity and importance of ML techniques in the analytics for prediction and forecasting are discussed. The chapter covers need of incremental approach in the analysis of the Big Data with applications to the same. Chapter 9 covers the aspects of data analytics to create value. It also covers some important aspects of business analytics. Chapter 10 concludes the discussion. It summarizes some important aspects covered in the book giving a few pointers for more thinking. Annexure 1 gives introduction to Hadoop framework from Big Data perspective.

◇ SUMMARY

Big Data has become the buzzword across the globe. Big Data is not just about huge site data but there is a significance to that size with reference to problems we are trying to solve. Big Data actually gives answers' to some of the questions related bottlenecks of traditional data mining. Big Data is not the change in technology, rather it is change in paradigm. Paradigm is related to holistic data analysis and decision-making. Hence, traditional machine learning methods do not work for it. Text analytics is a major part of Big Data. While we go through learning of unstructured data mining and text analytics, many subtle aspects of Big Data can be observed and we would have systematic way to approach this new paradigm.

2

Data Mining and Modelling

—Dr. Prachi Joshi
—Prof. Sheetal Sonawane
—Dr. Parag Kulkarni

◇ 2.1 INTRODUCTION

Over the years, there has been a revolution in the field of data mining. From corporate to venture capitalists, all are interested in converting the available data into knowledge so as to achieve maximum benefits; a potential knowledge that is able to analyze and determine the future! So, it is all about prediction, forecasting and estimation. Starting from investment plan to forecast, from determining weather conditions to appropriate job selection, it is all about data and information mining. Data mining and data analytics are all about making data into talk, converting it into knowledge with reference to context and making it ready to take decisions.

The mining process involves capturing of meaningful information and puts forth the analysis. Gaining an insight in the hidden patterns and extraction of this knowledge are indeed a challenging task. As the technology progresses, newer algorithms and approaches have evolved to cope up with the ever increasing data. These approaches target on determining the effectiveness of the available data to build predictive models.

Though researchers have come up with all new algorithms in mining, what required is appropriate data modelling. 'Is data modelling outdated?' Is the buzz that coming up? But practically speaking for BI, data modelling is essential. It captures all the perspectives!

We are aware of that data mining as a whole is responsible for extracting maximum hidden meaningful information. It deals with the aspect of determining the entire process of capturing the different views of the data and defining their relationships. More or less mining occurs as a process in data modelling.

A typical process of modelling is given in Figure 2.1.

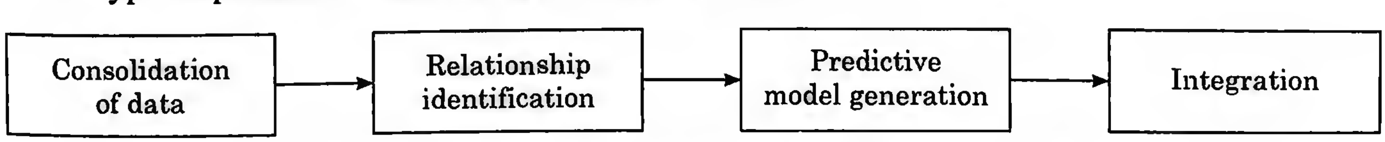


Figure 2.1 A generalized modelling process.

A generalized modelling process as shown in the figure is used to build models. This would differ with respect to the applications.

When we, in general, talk about mining, we are concentrating on understanding of useful patterns. Patterns that would help in giving an insight for forecast. Being familiar with the notion of what mining deals with and the various approaches available for mining, this chapter provides the reader a pathway to reach Big Data anaıytics. It is a journey from the modelling to mining and knowledge discovery to the future trends. Let us begin with the data models.

2.2 DATA MODELS

On broader scale of abstraction levels, we find the data model levels belonging to the following categories:

- Conceptual
- Logical
- Physical

Let us highlight mode on what these models mean. When we are discussing about the models, conceptual models deal and talk about the contents of the system. They essentially represent 'needs of the system'. So, to be point blank, it is all about the business requirements. These models explore and capture more of the business needs for the stakeholders.

Logical models are concern with the domains. They deal with the implementation. How necessarily the system would be implemented is looked upon here and the notion of database design is not accounted entirely. The model can be rightly said to be the one that considers the business needs and gets the implementation, though it would not be concentrating on the database structure.

Physical models are the ones that mention the database design and the details of the fields. So, they would be looking at both the aspects, that is, data base design and implementation.

Figure 2.2 shows the models.

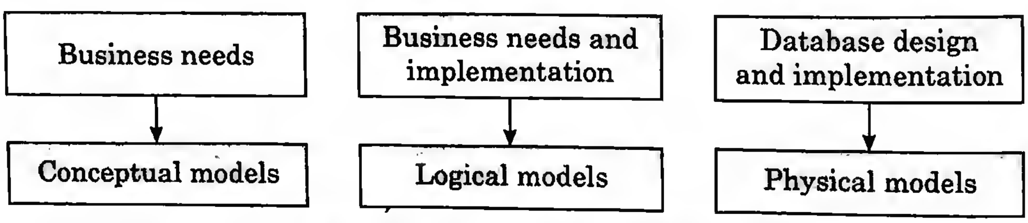


Figure 2.2 Data models.

Role of business analyst

For data modelling, we need to take decisions. For these decisions, data analysis is required. From the perspective of a business analyst, one would definitely be looking at the conceptual and the logical models. An analyst needs to look at the needs of the business and appropriately map into the model that would be used.

◇ 2.3 STAGES OF DATA MINING

We will focus on the data mining aspects in this section. Why and what is to be mined? Though, we say that modelling is an essential aspect, the analysis part deals with mining. What does it look at? At a broader scale, data mining can be categorized into three stages viz:

- Data preparation
- Model generation
- Deployment

Data preparation is sometimes referred as data pre-processing. It deals with cleaning, transformation and selection of the data. Prior to analysis, it is necessary that the potential features be identified that can yield effective outcomes from prediction perspective. This would be dependent on the nature of the problem at hand. The processed features will be the set where they would be showing relevance to the next stage of the analysis.

This stage needs to look at the transformations, the feature reductions and the normalization aspects for the data for it to be utilized in most correct way for the analysis.

The second stage of model generation is a phase that would deal with identifying suitable model. This involves recognizing the most promising model on the basis of their predictive evaluations available earlier. Here we would be looking at the various machine learning techniques and algorithms that would suit the problem at hand to get desired results. The toughest task is that, this phase deals with the selection of the best model, comparing their evaluation of performances.

Deployment is concerned with putting the model at work to use it for the predictive analysis. Figure 2.3 shows the details of the stages:

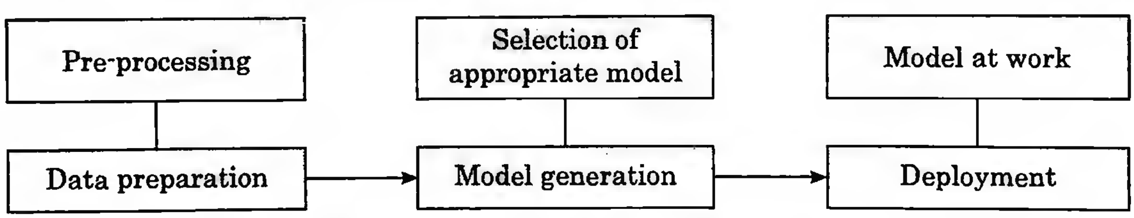


Figure 2.3 Stages in mining.

These are the basic stages to consider the data mining aspect. Let us move towards the knowledge discovery.

2.4 DATA MINING AND KNOWLEDGE DISCOVERY

It has been a point of debate over the years to establish the relationship between the knowledge discovery and data mining. Interestingly, it is more of discovery of the knowledge itself from the data. Knowledge discovery is extraction of previously unknown and interesting information from data.

To be more precise, data mining again is a step in knowledge discovery. The steps involved in knowledge discovery are depicted in Figure 2.4.

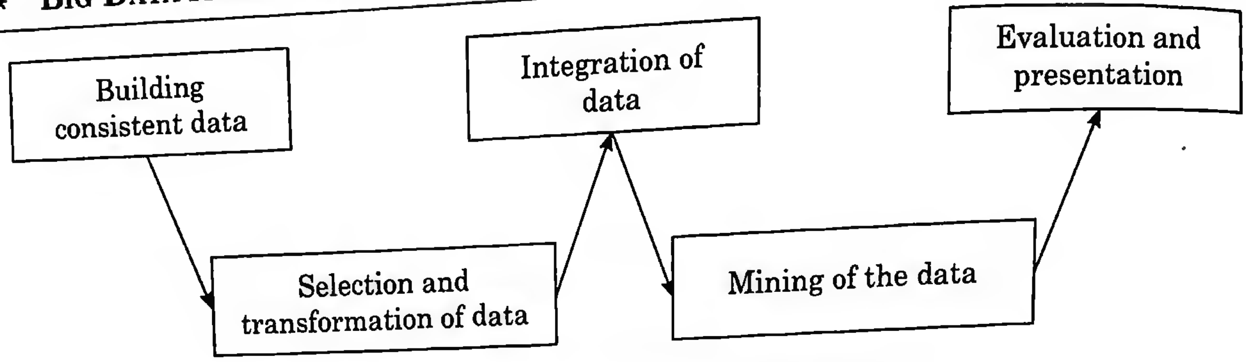


Figure 2.4 Knowledge discovery.

Knowledge Discovery from Data is often referred to as 'KDD', is the process that involves extracting, mapping, converting, transforming and selecting relevant data for applying and evaluating different mining approaches for effective analysis. It is a continuous process that should occur. One would find researchers referring to this process itself as data mining. The process is an elaborated one that is depicted in Figure 2.3. Let us deal with more detail over here.

- The first task involves representation of the data in a consistent way. This necessarily involves removal of the noise.
- The next step involves integration of the data. This deals with combining and formulating the data from multiple sources.
- Data selection and transformations on them are the further steps that look into the relevance of the data. This data is of immense importance from the mining perspective.

The above mentioned steps contribute in the pre-processing of the data prior to the mining activity. The further steps are as follows:

- Data mining which makes use of and applies various intelligent algorithms and machine learning techniques to retrieve meaningful data.
- The next step is the evaluation and presentation. Identifying and making available the appropriate information that is mined out is carried out in this stage.

◇ 2.5 ASPECTS OF DATA MINING

It is obviously clear that data mining is concerned with getting useful information from the data. But, mining on what kind of data? Various forms of data are available and mining needs to handle them. Mining approaches need to possess the potential to grasp and evolve with the new data as it emerges. Moreover, what sort of mining activity is required also needs to be identified. This section essentially describes the various mining aspects along with the data it deals with.

The Data

Today, we are getting acquainted with different forms and varieties of data. The data is growing enormously and is unstructured. It is turning out to be a Big Data. Mining is required today to handle this Big Data. The data sets dealt with mining are as follows.

- **Flat files:** It is the simple and most commonly available form of data that is used by mining systems. This type of data can include transactions or any other textual data.
- **Databases:** The mining systems are used in the analysis and prediction while treating the relational databases. They typically aim at discovering of patterns that can impact the growth factor or increase in sales for a product for example and so on.
- **Data warehouse:** Managing a multi-dimensional structure for mining is a challenging task. Mining activity here is concerned with exploration of varied combinations of data at different levels. It is dealt using an OLAP way.
- **Transactional data:** While dealing with transactional data, the mining system is more focused on association mining between the different set of data items.
- **Data streams:** Data transmission that takes place over a network or even the data from sensors that are being available continuously need analysis. Mining activity deals with these data sets as well. Many real time applications generate data in the form of data streams.
- **Spatial data:** Large amount of information can be mined from the spatial data as per the demands. By spatial data, we mean to say the maps that contribute in geographical information, or any positional details. Prediction activity is more often a point of focus in these datasets.
- **Multimedia data:** This data includes the images, videos, audios and even the text media. Mining out relevant information from such kind of data is a complex task and involves image processing, computer vision as well as natural language processing activities.
- **Time series data:** These datasets are often about the stock markets, user login information and so on. Thus, the mining activity involves real time analysis and has to capture the trends of the pattern.
- **World Wide Web:** It is an ever increasing and widely available data on the internet which is heterogeneous in nature. Mining data here is actually a combination of the previously listed data. Mining process over here is referred to as web mining.
- **Big Data:** It is a huge data that involves a species of data from text to images, audios to videos and any other combinations as well. More often it is observed as a stream of data. Big Data has actually opened up many challenges to the mining process. To manage the volume, velocity and variety, the mining approaches that are traditionally used are not sufficient. Use of parallel and scalable architecture to exploit mining activity is required as of now. The mining venture would change with the use of parallel processing and distributed storages, that is, where the future aims at!

What sort of mining?

While discussing the datasets that mining needs to look at, what sort of mining activities are possible are discussed in this section. We have the clarity that mining is able to deliver analysis that is useful for prediction, estimation or even forecasting. But it does not end here. There are many facets to it. Let us take it one by one:

- **Categorization/Classification:** This aspect of mining involves classification of the data. The classification is based on some learnt classes. Often referred to as a 'Supervised' Machine Learning approach, the classifier uses a training set to build its model. The built model is then used for classifying unknown sample. Figure 2.5 depicts the same. For example, set of documents belonging to category of personal loan (Class A) or home loan (Class B) forms a training set (labeled data). A classifier is built to understand this and generate a model. When an unknown document is provided to it, it should accurately classify this document to Class A or Class B. Many different classifiers are commonly put to practice here. For example, Naïve Bayes, Decision trees, Neural Networks and many more.

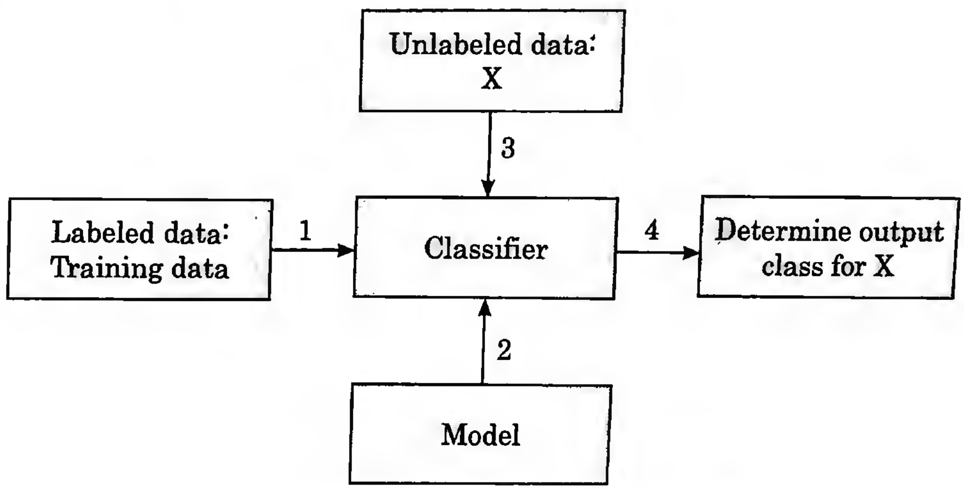


Figure 2.5 Supervised Learning.

While we have already discussed about the supervised approach being used for classification, the data mining here also performs regression. Regression is used to determine a numeric value. Thus, both classification and regression can be categorized to perform prediction.

- **Clustering:** We discussed about availability of the classes with supervised learning in the previous case, here the approach works with formation of groups of the data, which is unlabelled (data belong to a specific class is unknown). The grouping or clustering is also called as unsupervised learning. The clusters are formed based on the similarities, where the intra-objects are maximum co-related whereas inter classes are far apart. This approach can be made use of to actually assign labels for the groups formed. Figure 2.6 depicts a Cluster formation.

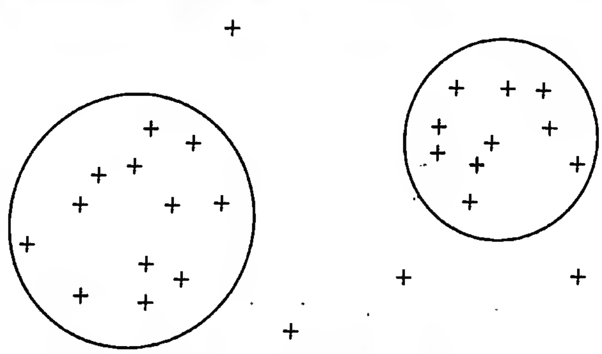


Figure 2.6 Cluster formation.

- **Characterization and discrimination:** This treats the basic operations of mining. The mining approach is concerned with summarization/determination of the class for a data which is based on the target class. For example, from given set of transactions occurring, finding characteristics, i.e., details of a customer who has invested at large in a specific stock. The possible output would be with respect to age, occupation and so on.

  In case of discrimination, a comparative study about these characteristics with respect to two or three different stocks can be generated. So, when we are discussing about characterization and discrimination, we are necessarily performing the operations of roll up and drill down of OLAP.
- **Associations:** It is a very interesting feature of mining. The association rule mining extracts and identifies frequent item sets. Referring to again a transaction data for a purchase of items, the association analysis identifies the relationship in the buying patterns for the items. For example, the relationship or the possibility that a customer is likely to buy a pair of socks while purchasing a shoe. These associations build association rules which help the shop owner in determining what items should be made available.
- **Outliers:** An important aspect that mining can take care of is outlier detection. By outliers we mean that determining an object/data that does not fit with the normal ones. They do not follow the normal behaviour and more or less are treated as anomaly. One would look different distance based measures in their detection though any of the supervised/unsupervised or even semi-supervised approaches can be made use of for the outlier detection.

◇ 2.6 DATA MINING APPROACHES

This section now deals with a few methods applied with examples for the mining purposes that are just discussed in the previous section.

◇ 2.6.1 Association Rule Mining

Nowadays, in various areas large amount of data is available on daily basis. For example, customer purchases data at grocery stores. Such data is called as market based transactions. Consider the typical example to understand the association rule mining.

In Table 2.1, each row denotes a transaction which contains a unique identifier labelled TID and a set of items purchased by a given customer. Sellers are interested in finding the behaviour of the customer. This important information is used to make business decision in various applications like marketing, advertisement, and CRM.

Here we are focussing on methodology called association analysis which is useful in finding interesting and useful relationships which are not easily analysed in large dataset. These kinds of relationships are represented in the form of generating frequent patterns or finding association rules.

Table 2.1 A sample super market transactions' data

TID	*Items*
1	{Bread, Eggs, Milk}
2	{Bread, Eggs, Cheese}
3	{Milk, White sugar, Bread}
4	{Bread, Butter, Cheese}
5	{Bread, Butter, Milk, Eggs}

For example, the following rule can be extracted from the given dataset

$$\text{Bread} \rightarrow \text{Milk}$$

This rule shows strong relationship between sale of bread and milk. The person, who buys bread, also buys milk and probability of occurrence of association among these two items is high. This association is helpful for retailer for selling their products to customers.

Other than market-based analysis, association analysis is useful in various applications like medical, web mining, information retrieval and bioinformatics.

There are following major challenges for association analysis for market based data:

- Large transaction data
- Identified pattern may mislead the analysis

Let us discuss the basic concepts and the algorithm in detail.

Problem statement

Market-based data can be represented in a graph format as shown in Figures 2.7(a) and (b), where adjacency matrix representation is shown. Rows and columns indicate an item. The number of times it occurs together is used in the matrix. The count itself shows the importance of its occurrence.

This representation is simple view of market data as it highlights the number of occurrences of the item together and easily helps to find frequent itemset.

Items	*Bread*	*Eggs*	*Milk*	*Cheese*	*White sugar*	*Butter*
Bread	–	3	3	2	–	1
Eggs	2		1	1	–	
Milk	3	2	–	–	1	1
Cheese	2	1		–	–	–
White Sugar	–	–	1	–	–	–
Butter	2	1	1	–	–	–

Figure 2.7(a) Adjacency matrix representation of sample data.

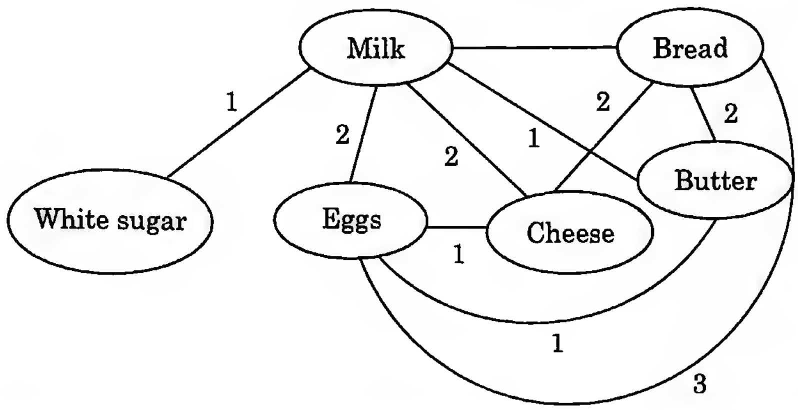


Figure 2.7(b) Graph representation of sample data.

Itemset: Let I be the collection of items in a basket data, then

If number of items in a itemset is l, then it is called l itemset. For example, {Bread, Butter, Milk} is an example of 3-itemset. The empty itemset contains no itemset.

Transaction: Let T be the collection of all transactions, then

$$T = \sum_{t=1}^{N} t_l$$

Each transaction t_i contains a subset of I.

$$t_i = \{t_{ij}, \text{ such that } t_{ij} \text{ is a subset of } I\}$$

For example transaction $T1$ = {Bread, Eggs, Milk}, where Bread is a subset of itemset I.

Support count: Support count refers to number of transactions that contain a particular itemset.

$$\text{Support count } (L) = |t_i, \text{ such that } L \text{ is a subset of } t_i \text{ and } t_i \text{ is subset of } T|$$

For example, support count of {Bread, Eggs, Milk} is 1. There is only one transaction that contains all three items.

Association rule

Association rule indicates the association between two itemsets. For example, $A \rightarrow B$, where A and B are disjoint sets. The power of association rule can be measured in terms of support and confidence. Support denotes the number of times both the itemsets available in a given dataset, whereas confidence denotes how frequently items in B available in transaction that containing A. The definitions are as follows:

$$\text{Support, } S(A \rightarrow B) = \frac{\text{Support count } (A \cup B)}{N}$$

$$\text{Confidence, } C(A \rightarrow B) = \frac{\text{Support count } (A \cup B)}{\text{Support count } A}$$

For example, consider the rule {Bread} $\rightarrow$ {Milk}. The support count of {Bread, Milk} is 3. The numbers of transactions are 5.

Hence,

$$\text{Support} = 3/5 = 0.6$$

and confidence is = support count of {Bread, Milk}/support count of {Bread}

$$= 3/4 = 0.75$$

Support and confidence are the important measures for business decisions. Support is a simple measure which proportionate the occurrence of itemset in transaction by number of transactions. If the confidence is high.

More likely for *B* to be present in the transactions that contain *X*, confidence is calculated using conditional probability.

Association rule mining

There are two major tasks to find association rule.

1. **Generate frequent itemset:** It is mainly used to find all itemsets which satisfy minimum support threshold.
2. **Association rule generation:** It is mainly used to extract all the high confidence rules from the frequent itemsets found in the previous step. These rules are powerful rules.

Apriori principle

Apriori principle is basically used to reduce the number of candidate itemsets found during frequent itemset generation. The principle says, 'if an itemset is frequent, all its subset must also be frequent. Figure 2.8 depicts a simple example of the same.

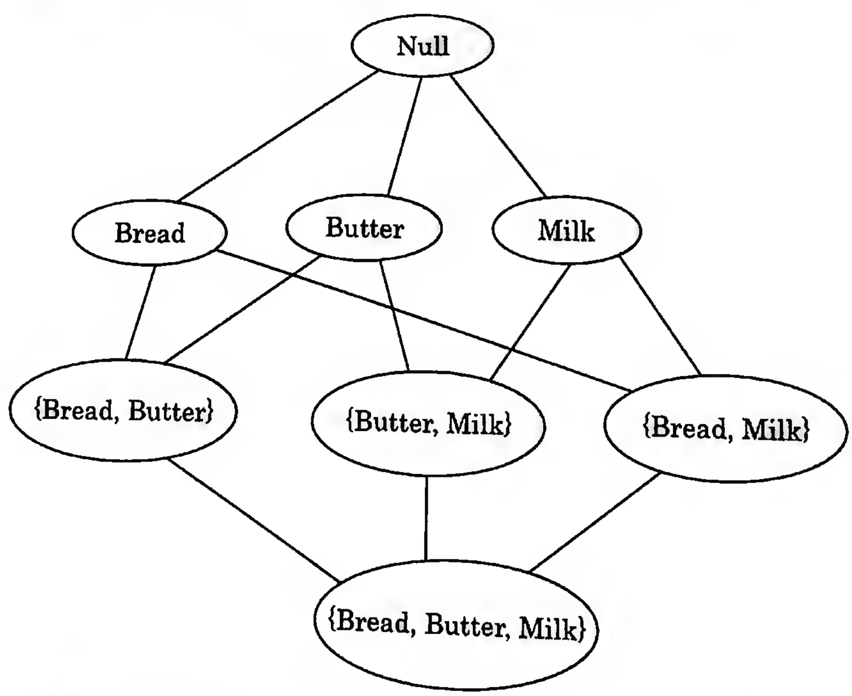


Figure 2.8 An example of Apriori principle.

For example, if {Bread, Butter, Milk} contains in transaction, then it is all subset, i.e. {Bread}, {Butter}, {Milk}, {Bread, Butter}, {Butter, Milk}, {Bread, Milk}, are also present in the transaction. As a result, all subset of a given set are also frequent.

Conversely, if an itemset is not frequent, all its supersets must be infrequent.

STEP I Frequent itemset generation using Apriori algorithm

We assume threshold support count = 2, which is equivalent to 40%.

Step 1:1 Frequent itemset

Item	*Count*
Bread	5
Butter	2
Eggs	3
Milk	3
Cheese	2
White sugar	1

In this step, every item is considered. We discard the item which is not satisfying minimum support count. For example, {White sugar} is discarded for the next step computation.

Step 2:2 Frequent itemset

Item	*Count*
{Bread, Butter}	2
{Bread, Eggs}	3
{Bread, Milk}	3
{Bread, Cheese}	2
{Butter, Eggs}	1
{Butter, Milk}	1
{Butter, Cheese}	1
{Eggs, Milk}	2
{Eggs, Cheese}	1
{Milk, Cheese}	0

Number of possible candidate 2-itemset are $\binom{5}{2}$ are 10. We found that out of 10 candidates itemsets 5 are frequent.

We found that out of 6 candidate itemsets, only 1 is frequent. Hence {Bread, Eggs, Milk} is frequent candidate 3 itemset.

Step 3:3 Frequent itemset

Item	*Count*
{Bread, Butter, Eggs}	1
{Bread, Butter, Milk}	1
{Bread, Butter, Cheese}	1
{Bread, Eggs, Milk}	2
{Bread, Eggs, Cheese}	1
{Bread, Milk, Cheese}	0

Algorithm steps for generating frequent itemsets

STEP 1 Find support of each item. The algorithm needs to make an additional pass over the data set.

STEP 2 Find the set of all frequent 1-itemsets. The algorithm eliminates all candidate itemsets whose support counts are less than minimum support threshold.

STEP 3 Generate iteratively k-itemsets using the frequent $(k - 1)$ itemsets generated in the previous step. Simple "Join" operation can be used to generate candidate itemset.

STEP 4 Stop when no new frequent itemsets generated.

Association Rule Generation

This section elaborates to extract association rules efficiently from a given set of frequent itemsets. Using above dataset and algorithm of frequent itemset generation, {Bread, Eggs, Milk} is frequent 3-itemset.

So, possible association rules are as follows:

Bread, Eggs → Milk
Bread, Milk → Eggs
Eggs, Milk → Bread
Bread → Eggs, Milk
Eggs → Bread, Milk
Milk → Bread, Eggs

These generated candidate set should satisfy minimum support value and confidence value. Confidence of rule {Eggs} → {Bread, Milk} is calculated using support count {Bread, Milk, Eggs}/support count {Eggs}.

Merits and demerits of Apriori algorithm

Apriori is one of the most popular and successful algorithm for generating frequent itemsets. The search space is reduced by Apriori principle. But the algorithm still has I/O overhead since it requires large number of passes over transaction dataset. Hence, the performance may degrade in case of large dataset. Various algorithms are addressed to handle this issue namely, FP growth which makes use of hashing mechanism for reducing number of passes.

Applications

Association rule are applicable in various application domains like information retrieval, text mining, web mining, network intrusion detection and bioinformatics. The association rules have also used in different mining task like classification and clustering.

◇ 2.6.2 Naïve Bayes

Naïve Bayes is one of the most popular supervised machine learning approach. It is a probabilistic classifier. It works on the Bayes theorem that exhibits independence assumptions among the variables involved. It is often used in classification of text, mails and information filtering applications. It works well with comparatively small amount of training data. Let us begin with the Bayes theorem.

Bayes theorem is used to calculate posterior probability that the hypothesis holds. This hypothesis calculation is based on (i) prior probabilities—referred to as known values of probabilities, (ii) probability of observing various samples of data given the hypothesis and (iii) probability of the observed data under the absence/no knowledge of the hypothesis.

Consider $P(h)$ to be the initial probability that hypothesis holds. That is the background knowledge that h is correct prior to the availability of the training dataset.

$P(d)$ to be the probability of d (training data). This has no knowledge about h.

$P(d|h)$ to be the probability of observing d with some given hypothesis h. (read as probability of d given h).

$P(h|d)$ to be probability that hypothesis h holds given the training data d—posterior probability of h (read as probability of h given d).

To calculate the posterior probability, the formula is:

$$P(h \mid d)\frac{P(d \mid h) \times P(h)}{P(d)}$$

Since we are saying that it belongs to the category of supervised learning, we are essentially trying to predict the class. When we say that we are performing prediction, the hypothesis here stands for the classes the data would belong to.

Let us take a simple example to understand the concept. Assume that you have to predict the probability that a player X gets selected in a team. So, here the classes for h are Yes/No. So, given a player X, we intend to find 'what is the probability that he/she gets selected in the team'. This is the posterior probability: $P(h|d)$, where $h = Y/N$.

Let us take one detail example that will show us how the classification works. Let us assume that there are three sets of classes, Teaching Assistants (TA), Research Associates (RA) and others.

Table 2.2 shows training samples of students belonging to each of the classes as available:

From the data available for 9 total cases, let us build a probability table. Table 2.3 is simply computing the occurrence of the values in the training data. For example, under the paper publications column we have put two classes: TA and RA. In TA for P published class, the value is 1 indicating there is only one sample available that belongs to TA class and has published paper. Similarly other values are filled.

Table 2.2 **Training dataset**

Class	*Paper publications (P/S/N)*	*Co-curricular (P/NP)*	*Sports (I/N/NP)*
TA	Nil	Participated	International
RA	Published	Not participated	National
TA	Submitted	Participated	National
TA	Published	Not participated	Not participated
RA	Submitted	Not participated	International
RA	Submitted	Participated	International
RA	Published	Not participated	Not participated
RA	Submitted	Participated	Not participated
TA	Submitted	Not participated	Not participated

Table 2.3 **Occurrences of every attribute with respect to the class**

Paper publications			*Co-curricular*			*Sports*			*Class*	
–	*TA*	*RA*	–	*TA*	*RA*	–	*TA*	*RA*	*TA*	*RA*
P	1	2	P	2	2	I	1	2	4	5
S	2	3	NP	2	3	N	1	1	–	–
N	1	0	–	–	–	NP	2	2	–	–

Now, let us compute the probability values:

Table 2.4 **Probability value calculations**

Paper publications			*Co-curricular*			*Sports*			*Class*	
–	*TA*	*RA*	–	*TA*	*RA*	–	*TA*	*RA*	*TA*	*RA*
P	1/4	2/5	P	2/4	2/5	I	1/4	2/5	4/9	5/9
S	2/4	3/5	NP	2/4	3/5	N	1/4	1/5	–	–
N	1/4	0/5	–	–	–	NP	2/4	2/5	–	–

What do these values indicate?

Let us take the value of Paper Publications, TA column and *P* row. The value here is 1/4. This 1/4 tells us the probability *P*(Paper publications = Published | TA class). That is probability that the publication is published, given in the TA class.

This is $P(d|h)$.

Similarly, other values are computed.

What is $P(h)$? $P(TA) = 4/9$ and $P(RA) = 5/9$.

Since no information about d is available, this factor is ignored.

Now, given an unknown sample X with following values

Paper publications = Published, Co-curricular = Participated, Sports = National

What is the probability that it would be classified in TA class or RA class?

Let us do the calculations:

P(TA | X) = *P* (Paper publications = Published | TA) * *P* (Co-curricular = Participated | TA) * *P*(Sports = National | TA) * *P*(TA)

= 1/4 * 2/4 * 1/4 * 4/9 = 0.25 * 0.5 * 0.25 * 0.44 = 0.01375

Similarly, we calculate *P(RA | X)* = *P* (Paper publications = Published | RA) * *P*(Co-curricular = Participated | RA) * *P*(Sports = National | RA) * P(RA)

= 2/5 * 2/5 * 1/5 * 5/9 = 0.4 * 0.4 * 0.2* 0.55 = 0.0176

So, the probability of RA is high, and hence it is classified into the RA class.

This is a simple example to understand how the approach works. The values taken in the example are categorical ones. The approach works with continuous values as well. In such cases, Gaussian distribution is used.

Queries?

1. What will happen if the values are 0? Like in the above example, the value for the RA and no paper publication was zero. Under such situations, a technique called smoothing is used. The zero value is converted to non-zero by considering additional data sample.
2. Does Naïve Bayes guarantee to give correct output? Even if the probability values for known data samples are low, the approach shows good classification results. It is preferred as it is computationally less expensive. Though there are many other classifiers which have outperformed Naïve Bayes.

This was one of the supervised approaches. There are many others like SVM, decision trees, ensemble methods, neural networks and many more.

◇ 2.6.3 *k*-means Clustering

In the previous section, we dealt with one supervised approach. In this section, we discuss one unsupervised approach and the simplest one, *k*-means.

In the earlier sections, we discussed in brief on the unsupervised approaches. Essentially, they belong to different categories like partitioning based, hierarchy based, density based and grid based. The approach of *k*-means belongs to partitioning based.

Let us understand the working of *k*-means.

Unlabeled data: Data whose classes are unknown, the approach clusters/groups them based on the similarities between the objects. It partitions them. The most common 'Distance' rule used here is the Euclidean Distance.

How does *k*-means work?

Let us understand the algorithm:

1. Input: (i) The data points
   (ii) Number of clusters = *k*

2. Select centroids or seed points at random, from the given data points, which are equal to k.
3. For every data point x:
   (i) Calculate the distance between x and the centroids.
   (ii) Assign the point to the closest centroid (i.e., it is assigned to the cluster represented by that centroid). 276397
4. Set the position of the cluster centroids, now to be the mean of the all the data points assigned to that cluster.
5. Repeat Steps 3 and 4 till the convergence.

From the algorithm discussed, it converges indicating that there is no movement of the data points in the subsequent iterations from one cluster to another.

k-means is a technique that tends to minimize the distance between the data objects within the cluster. Though the approach can be used for a wide range of applications, the approach needs accurate selection of k value. That is the performance will affect with an inappropriate selection of the value for k. There are different ways to have this selection like one would be working with different values of k and then finalizing k number of clusters where the error is minimum!

To add further, the approach also faces the problem of dependency on the initial centroid selection. The convergence factor is dependent on this.

There have been many improvements suggested and are in practice for k-means. One of the most common known variant is k-medoids.

CASE STUDY: Turning Data into Business Value

Mr. Satish Gandhi used to collect various batches of raw chana and used to analyze those batches to log in different properties of those chana. These include size of chana (chickpeas), the thickness of skin, number of projections, average weight, skin colour etc. he continued the study and kept on logging parameters and also discovered a few more parameters. The data kept on growing and more and more features were accumulated. In 2010, he took a decision to go systematically to mine this data, classify this data and map it for quality control. He used ensemble classifiers to classify this data. For that, he did very careful feature engineering and selected 35 most important features to classify raw chana. In the process, he did some tuning and carefully weighted all features. With his efforts and application of rough set theory along with ensemble machine learning, he classified raw chana into 4 classes. Class one is the best quality and gives the best quality outcome in one cycle. Other classes are: grade two, grade three and grade four. This improved his overall quality of output resulting in more than 99 percent accuracy. This helped him to crack major deal and today they are exporting chana to more than 40 countries. Thus, he converted data into business value to improve the quality and overall market penetration.

(Ref: *Knowledge Innovation Strategy*, by Parag Kulkarni, Bloomsburry India)

◇ 2.7 CRAWLING THE WEB AND INFORMATION RETRIEVAL

We now look at the Web Crawling aspect and Information Retrieval (IR). Every now and then

when we need any information, we seek it from the search engines. There has been a significant growth for the usage of the web for these searches. This has given rise to effective searching of rich web contents. Web searching essentially deals with searching web documents. Where information retrieval is the field where user extracts required information from large collection of text documents, web search is an application of information retrieval. No, the information to be extracted should be relevant to the query user has fired. There is 'relevance score' that is calculated to the given query. Further, a ranking is performed on the basis of relevance score. Though web search is a big thing that is happening now and has to efficient in terms of speed and delivering relevant contents, it faces following major challenges in the retrieval systems:

- Managing large and increasing collection of web document
- Volume of user query that is supplied on a daily basis
- Extending use of searching for other use such as advertising and recommendation system
- Today's need of e-commerce

IR systems are continuously trying to cope up with these issues by indexing and ranking. Data retrieval systems work on relational database where data is structured while information retrieval systems are more focused and diverted to work on natural language text where data is unstructured. One more point of difference between data retrieval system and IR is that data retrieval systems find a solution to the user of a database system, whereas IR systems find a solution of retrieving information about a topic or subject.

Simple architecture of IR system is described in Figure 2.9. Let us understand its working. The system begins with a collection of web documents by the web crawler. This document collection is stored in a document repository. In order to avail fast retrieval and search, the documents are indexed. An input query is provided by the user to search the required information. The query is parsed and processed against indexed document collection, and documents are retrieved. At the last stage, the documents are ranked, and top documents are returned.

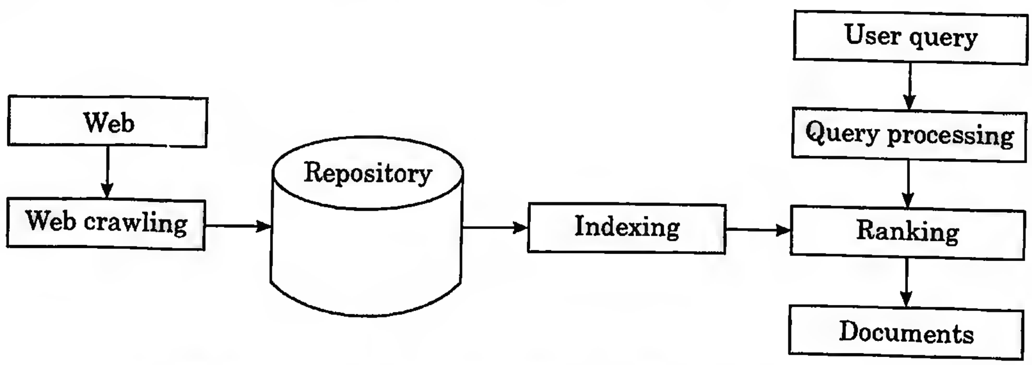


Figure 2.9 Architecture of IR system.

Let us put it mathematically:

Let system $S = \{D, q, \text{Ranking function}, O\}$

where D is the collection of web documents which is represented as:

$$D = \sum_{i=1}^{n} d_i$$

Each document is a collection of terms d_i represented as:

$$d_i = \sum_{j=1}^{m} t_{ij}$$

The input query q is collection of query terms given as:

$$q = \sum_{i=1}^{t} q_i$$

And O is the output list of ranked documents.
The ranking function maps the query q to the document collection D.
F (Ranking function): $q \in D$ and $O \in D$
That is how an IR system works, but now we will deal in details about the web crawling.

◇ 2.7.1 Web Crawler

Web crawler is the first component in IR architecture. In simple terms, web crawling means collecting pages from the web so that it is useful for ranking. The goal of web crawler is to efficiently collect web pages. But while doing so, the crawlers need to ensure that the established links are also maintained and preserved. They are usually called spider or robot.

The fundamental algorithm is simple. It takes seed URL as input. Crawler downloads all the web pages which are addressed by URLs. It also extracts all hyperlinks contained in the pages and downloads these web pages.

Following challenges are faced by web crawlers:

- Large and increasing collection of web that leads to complexities in the collection process.
- Selection of useful sites or prioritise the crawling process becomes complicated with this growth of the web data.
- Detection and handling misleading sites are tough jobs often encountered by the crawlers.

Let us look at the architecture of the web crawlers.

Figure 2.10 shows basic crawler architecture. However, this crawler extracts one page at a time. The efficiency of crawler can be improved by the use of multiple threads or processes.

Let us discuss the working of the crawlers. Queue data structure is used in main memory for keeping URLs of unvisited page. The seed URLs are the collection of unvisited URLs which are specified by the user. The crawler takes URL from queue, extracts the page, parses the page to extract its URLs and adds new URL to queue. It stores retrieved page to local repository. The crawling process continues till queue is empty or forced to stop.

Web crawlers can be implemented as distributed crawler to increase the throughput. This can be done dividing the URL space. URL space is partitioned across web sites.

Web crawlers can also be used to perform incremental crawling to discover newly found pages and recollecting previously crawled pages. To achieve this, changes in the data structures are expected such as prioritizing the visited URLs and monitoring the web pages for temporal behaviour.

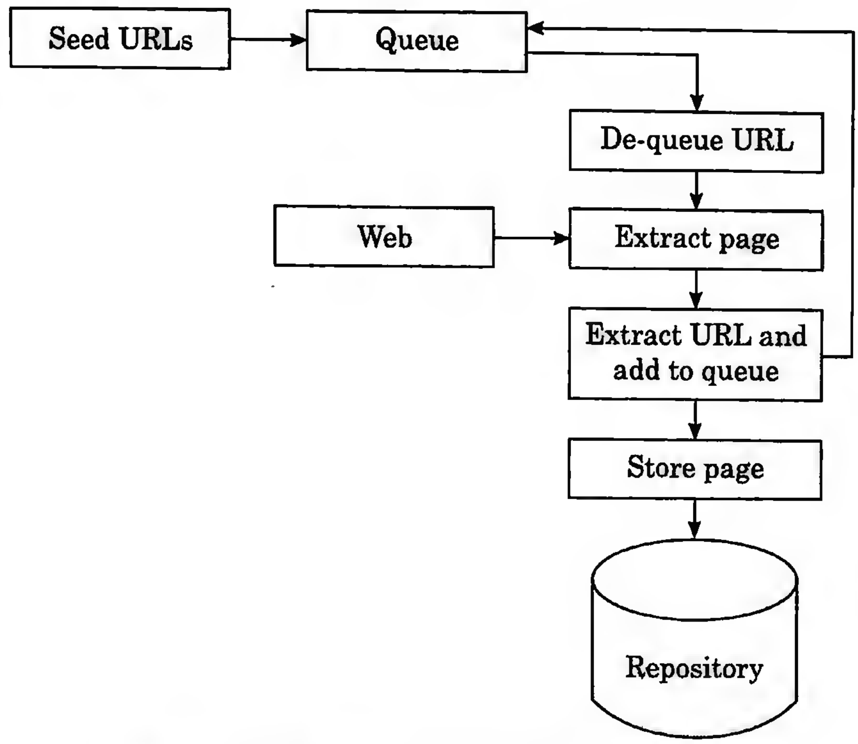


Figure 2.10 Crawler architecture.

◇ 2.8 RECOMMENDER SYSTEMS

We now highlight one of the important characteristics of data mining—Recommender System. Recommender Systems are considered to be a sub topic for the 'Information Filtering'. These systems tend to predict a preference and recommend one. They are able to justify based on the knowledge, and they generate and suggest a suitable output. These systems tend to assist the user in deciding preferences by considerable reduction in search and navigation.

They use data mining techniques to give their recommendations based on the learnt knowledge that is acquired on the basis of the actions and the characteristic features for the user. Recommender systems can use classification, clustering or even association rule mining to present their output. But the most commonly used technique is the association rules.

Three types of Recommender Systems (RS) found in literature include:

- Rule based
- Content based
- Collaborative

Rule-based systems make use of traditional filtering techniques. They perform the typical information search and retrieve an output that is recommended.

Content-based techniques work on the user preferences in the past. So, user profiling is taken into consideration. Representation of referred items is in terms of keywords here. But the system has many drawbacks:

1. The representation cannot deal with all the objects.
2. It may not be possible for all the items to be represented properly.
3. In case of multiple items/objects buying pattern by a user, they are unable to handle it properly.

Collaborative filtering techniques are the ones which are gaining immense importance nowaday, owing to the enormous usage of the social networking sites. These RS rely and use other user's rating for an object. They happen to discover and identify objects that would interest user based on other user's profile. Typical example of this is a recommendation for X persons on your social network site. Though the approach seems to have potential for building a strong recommender system, it faces few issues. These are as follows:

1. The sparse data is a major concern for these recommendation systems. They often face a problem of 'cold start'. A user who has given a newly available item and wants the RS to give its output, the RS could be ineffective owing to the lack of the data available earlier. May be very few users would have rated it and that information is just not sufficient enough for RS to operate.
2. Synonym usage for an item is also a major hindrance in the efficiency of the RS. The systems can treat the same item as different items and hence fail to give the expected result.
3. At one side where there is a scarcity of data for a specific item, there is a case where the data is growing continuously. This is in terms of the users and the new items. The RS systems need to cope up with this increase.
4. A very well-known issue is 'Gray Sheep'. These are the users whose opinions or rating do not belong to any one group for agreement. Thus, the collaborative RS fails to reap the benefits of accurate decision-making.
5. A very interesting problem that arises in collaborative filtering is Shilling attack. It could be a case where users tend to give good opinion on the items that are of their own company made/relatives, or there is some influence. They would in these scenario mention negative ratings for other competitors. In these cases, the collaborative filtering is unable to produce correct recommendations.

Very recently, study shows that use of hybrid mechanism for the systems makes use of content and collaborative technique as well. These to some extend can address the issue of cold start and sparse data.

◇ 2.9 CURRENT TRENDS

Data mining is the process of knowledge discovery where knowledge is achieved by analyzing the data in large collection. Data is analyzed and processed into useful information.

Due to increase use of various functionalities in different areas, it has become very important field. The use of machine learning, artificial intelligence and pattern recognition have raised its height. Data mining is useful in various applications from business, education, medical to scientific. Figure 2.11 covers the current trends of mining.

The improvement and need in various application fields posed new challenges to data mining. The challenges include:

1. Data diversity
2. Advancement in computing and networking resources
3. Different formats of data

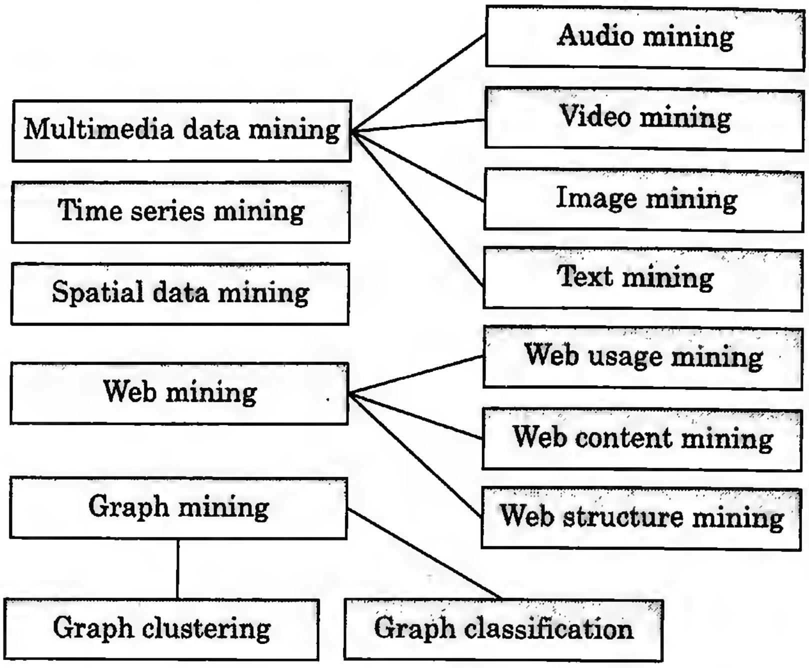


Figure 2.11 Current trends in data mining.

◇ 2.10 WHERE DOES THE FUTURE LIE?

Let us understand what facets are coming up for data mining.

Distributed Data Mining (DDM)

The goal of distributed data mining is to effectively mine distributed data which is located in heterogeneous sites. Examples of this include biological information located in different databases, data which comes from the databases of two different firms, or analysis of data from different branches of a corporation. Combining these data is an expensive venture as well as time consuming.

Distributed data mining is used to offer a different approach to the traditional approaches for analysis. They make use of a combination of localized data analysis, together with a global data model.

Ubiquitous Data Mining (UDM)

The advent of laptops, palmtops and cell phones is making ubiquitous access to large quantity of data possible. Advanced analysis of data for extracting useful knowledge is the next natural step in the world of ubiquitous computing. Accessing and analyzing data from a ubiquitous computing device offer many challenges. This arises owing to the fact that UDM introduces additional cost due to communication, computation, security, and other factors. So, one of the objectives of UDM is to mine data while minimizing the cost of ubiquitous presence.

◇ SUMMARY

Data Mining is all about converting raw data coming in different format into useful information and knowledge. Since there is huge data coming in different formats and different sources data mining performs different tasks right from data aggregation to classification of data.

There are different data mining models. This chapter has covered different models and types of data mining. We have also discussed different standard algorithms used for data mining. In subsequent chapters, we will cover different challenges of mining unstructured data.

Multiple Choice Questions

1. In a class, a camera captures photos of students at the start of a scheduled lecture and at the end. To determine presence of a student, which method among the following is most appropriate?
   (a) Supervised learning
   (b) Clustering
   (c) Association mining
   (d) All of the above
   (e) None of the above
2. Which of the following is drawback of Apriori algorithm is-
   (i) Time consuming due to support value calculation
   (ii) Slow owing to bottleneck in Candidate generation
   (iii) Performance degrades due to large dataset
   (a) (i)
   (b) (i) (ii)
   (c) (ii) (iii)
   (d) All of the above
3. The Class prediction in Naïve Bayes is decided with which value
   (a) Maximum posterior probability
   (b) Maximum value of the hypothesis
   (c) Maximum prior probability
   (d) Likelihood value
4. Which of the following is true about Naïve Bayes?
   (i) Can incrementally update prior probabilities and likelihood with new samples
   (ii) Works with discrete and continuous values
   (iii) Conditional independence assumption
   (a) (i) (ii)
   (b) (ii) (iii)
   (c) (iii) (i)
   (d) All of the above

Concept Review Questions

1. Discuss the data mining stages and role of knowledge discovery in the mining.
2. Differentiate between association mining and classification.
3. Explain the Apriori principle.
4. With an example explain Naïve Bayes
5. Explain the architecture of an IR system.

Critical Thinking Questions

1. Can Association mining used to determine whether there is a possibility that a novel written by an author will be liked by readers.
2. If you were to select a college to get admission for higher studies, what sort of machine learning methodology would you consider? Justify.

Lab Assignments

1. Using Naïve Bayes determine the probability of possibility of a candidate getting elected in the elections. Generate labelled data for attributes like: Education, Party, Years involved in Social service and so on.
2. Implement k-means clustering to group students to form project groups. Assume suitable parameters for formation of the same.

3

Big Data Mining
Application Perspective

—Dr. Sarang Joshi

With rapid growth in technology and markets, the Big Data is being generated due to huge number of transactions. To discover the knowledge, it is very important to process this Big Data nowadays. Conventional SQL method is very time consuming and may work with very large outcome tables and hence it may result in very slow decision system using knowledge discovery. To overcome this problem, integration of data mining and Big Data may be a key solution. This chapter conceptually discusses the integration of Big Data, along with the data mining methods and patterns. The examples and illustrations are given using MongoDB, a FOSS derivation of Big Data.

◇ 3.1 INTRODUCTION

The term 'data' came to wide existence with introduction to relational database management systems, popularly known as RDBMS. The Structured Query Language (SQL) is used widely to perform number of database operations. The column based structured information with complexity of 'WHERE' clause and JOINS was the common practice. Being relational in nature, large data with very few alpha-numeric columns was the nature of an output display. This may be searching the data and composing the reports. Addition of functions associated with the RDBMS can help in generating decisions based on data or data manipulation.

With rapid growth in computing, storage, GUI and Interface technology, the data paradigm became more inclusive of different formats of the information such as images, video, blogs, twits and multilingual in nature. For example, the daily news paper has different types of information which includes text news, photographs, images, advertisements, cartoons, etc. which use different formats of data representation. The e-paper visualization demands database to support different media and format requirements. Such requirements gave birth to reuse of the old concept, called Big Data. The programming language such as SQL (Structured Query

Language) also revised to NoSQL (Not Only SQL). The Big Data contributes in better time and space complexity by using divide-and-conquer of data to meaningful sub-divided data. The Big Data is characterized by following features; the traditional relational database systems fails to respond or poorly responds to these characteristics.

- The volume of data, usually 5 Petabytes and above size of data is considered as a Big Data.
- The rapidness or velocity at which data is collected or gathered is very large. Typically, the data crowding occurs on social websites like Twitter and Facebook by means of multiple users commenting on a twit or a Facebook message. Another example can be time-based data collection system such as temperature monitoring system and CCTV security systems.
- The variations of the media used are third characteristics. For example, typically, the data crowding occurs on social websites like Twitter and Facebook by means of multiple users commenting on a twit or a Facebook message using text messages, videos, audio messages, etc. The data crowding due to devices like RFID and such sensor networks is also a candidate Big Data.
- The outcomes such as text columns, numerical, graphs, images, audio, video, blogs and tweets can be generated by Big Data as against columns outputs generated by relational database.

The Big Data is an umbrella of structured, semi-structured or non-structured data, as against only structured data used by the relational database.

- Big Data contributes in a knowledge discovery from data from very massive unstructured data. It is also called Big Data mining whereas conventional data mining is about knowledge discovery of given object.
- Big Data is generated by heavy data-subdivision so that meaningful results can be obtained.

Different Big Data storage technology solutions are available in the technology market; MongoDB is an open source Big Data with NoSQL support. Pymongo is a popular and most convenient programming Python interface with MongoDB. Hadoop Distributed File System (HDFS) is one of the well-known solutions used worldwide to exploit Big Data. Typically, the Oracle enterprise Big Data solution includes a mix of open source and Big Data software such as Cloudera with Apache Hadoop-CDH4 with Admin manager, Oracle Manager, statistical package R, Oracle NoSQL community edition and Oracle Enterprise Linux Operating System with Oracle Java VM.

◇ 3.2 BIG DATA MINING

Data mining is a term used for discovering interesting details, patterns from massive storage of data. In other words, it is a knowledge discovery process. Any discovery process normally goes through number of steps, viz. cleaning of raw data, sorting or categorizing, processing to identify the data or discovery of data, protection and security, and storage.

The knowledge is discovered means different processes are applied using data properties, direct or indirect behaviour so that the required data becomes visible. Most of the times the data having direct visibility is applied with new dimension or formula related to the property or behaviour thus can be categorized which may lead to the knowledge discovery. All such cases are candidated for the data mining. Big Data having characterized with volume, velocity and variance has a huge scope of challenges leading to knowledge discovery. Business Intelligence is the best example the success of which heavily relies on knowledge discovery in Big Data.

Figure 3.1 illustrates the start of MongoDB service. The Big Data created using MongoDB can be used for variety of analysis using multi-dimensional data using text, images, audio, video and numerical or such datasets or itemsets. Figures 3.2 and 3.3 illustrate the creation of database.

```
soham@dhcppc0:~                                   x
File  Edit  View  Search  Terminal  Help
[soham@dhcppc0 ~]$ service start mongod
[soham@dhcppc0 ~]$ mongo
MongoDB shell version: 2.6.5
connecting to: test
>
```

Figure 3.1 MongoDB service illustration.

```
soham@dhcppc0:~                                   X
File Edit View Search Terminal Help
[soham@dhcppc0 ~]$ service start mongod
[soham@dhcppc0 ~]$ mongo
MongoDB shell version: 2.6.5
connecting to: test
> use Cust_data
switched to db Cust_data
> db
Cust_data
```

Figure 3.2 MongoDB use database illustration.

```
> db.Cust_data.insert({"Name":"Cust Name","SocialWebSite":"FacesBook","LoG":"AccessTDate"})
WriteResult({ "nInserted" : 1 })
> db.Cust_data.find()
{ "_id" : ObjectId("5484124f8c741daeba549f2f"), "Name" : "Cust Name", "SocialWebSite" : "FacesBook", "LoG" : "A
ccessTDate" }
>
```

Figure 3.3 Illustration of creating data entry in the MongoDB.

The MongoDB is a Big Data that has the ability to store, in addition to text and numerical, it can store multimedia objects. This is illustrated in Figure 3.4.

```
[root@dhcppc0 soham]# mongofiles -d database put /Home/soham/Music/rec.amr
connected to: 127.0.0.1
assertion: 10012 file doesn't exist
[root@dhcppc0 soham]# mongofiles -d database put rec.amr
connected to: 127.0.0.1
added file: { _id: ObjectId('54893138cd8fd60c3b002707'), filename: "rec.amr", ch
unkSize: 261120, uploadDate: new Date(1418277176729), md5: "be7c91ca0d7b6ba0e47e
a9a4c9be2665", length: 8998 }
done!
[root@dhcppc0 soham]# history >> mongo_multimedia
[root@dhcppc0 soham]# mongo
MongoDB shell version: 2.6.5
connecting to: test
> use database
switched to db database
> db.database.find().pretty()
> db.database.find()
> exit
bye
[root@dhcppc0 soham]# mongofiles -d database get rec.amr
connected to: 127.0.0.1
done write to: rec.amr
[root@dhcppc0 soham]# mongofiles -d database get 1.png
connected to: 127.0.0.1
done write to: 1.png
[root@dhcppc0 soham]#
```

Figure 3.4 Illustration of saving sound and image files to MongoDB.

The Big Data has the ability to store voluminous data with greater velocity along with variant data types. This feature of three Vs (Volume, Velocity and Variant data types) adds to the value during data mining. The knowledge generation and use of such derived knowledge

are the key features of data mining. The data received in the database is a raw database and requires cleaning based on the objectives decided for the expected or estimated outcomes.

◇ 3.2.1 Data Cleaning

Data cleaning is a process that accepts raw data which has lots of impurities and other redundant data with respect to the objectives of the query. The storage of raw data may result in occupying storage space due to impurities and redundant data which results into overheads of cost. The raw data may consist of redundant data, inconsistent data, errors or noisy and incomplete data. The data cleaning process identifies such redundancies and faulty data thereby improving the quality of the data. For example, data typing mistakes can be noted during data cleaning process. The massively large quantity is the major feature of a Big Data. Tools like IBM's SPSS and Stata (statistics/data analysis) can be used for cleaning, sorting and concatenating the Big Data. Data can be subdivided and processed using distributed concurrent environment for better time efficiency.

The data quality can be assessed using parameters like completeness, accuracy, consistency, time relativity, trusted data and interpretability of the data. The accuracy, consistency and completeness of the data majority depends upon the avoidance of the human errors like data entry errors and calibration of the devices used for capturing the Big Data. Human ignorance to complete all required data fields, relying on default values, also known as disguised missing data and data entry interpretation errors may result into errors and redundancies. Since the volume and velocity of the data captured are usually continuous with time, huge volume of data is captured with the speed. This may result into lot of redundant data, hence Big Data cleaning is one of the challenging tasks.

Time relativity of the data is yet another important reason of data redundancy. For example, inability due to various human tendencies resulting in delays for the timely data entry of say monthly sale or attendance of the students results into execution of timely processes without complete data results into voluminous collection of redundant data. This results into data cleaning algorithms regarding the trusted data. In other words, quality of data acceptance to be tested with the trusted data using techniques such as data templates, thresholding, calculation of mean and variance, etc. Such untrusted data may generate wrong or misguided interpretation, affecting the business decisions or losses. Hence, the data cleaning is very important step because without data cleaning data-garbage in huge quantity will occur resulting in performance of time and space efficiency.

Data reduction is one more intelligent method to identify the redundancies in Big Data. The performance of this reduced data is maintained closely equal to or exactly equal to the actual data. In other words, the data efficiency is maintained even though the data is reduced. The data reduction can be done by dimensional reduction or by numerosity reduction. Typically, image and video data use both the techniques. Since variety is one of the characteristics of Big Data, it becomes very essential to create a data-structure header to inform how the data reduction is done. This makes the data portable and light-weight.

The dimensional reduction methods use data encoding techniques to reduce or compress the cleaned data. Discrete Cosine Transform (DCT), Run Length Encoding (RLE) or, in general,

Principal Component Analysis (PCA), Wavelet transform are a few effective methods used for dimensional reduction of the Big Data. Also, small sets of attribute clusters are formed using the attribute construction methods, whereas attribute subset methods identify the priority of the attributes and dynamically select the redundant attributes for the elimination.

The numerosity reduction methods use replacement of data by small, light-weight data with the help of techniques such as parametric models. This includes regression or colinear models. Other techniques are non-parametric models such as sampling, histogram, clusters or data-aggregation.

◇ 3.2.2 Sorting and Categorizing the Data

The Big Data being very huge in size, it is very important to store it efficiently so that faster and timely retrieval can be possible. The data can be sorted provided it can be ordered using available criteria such as time-stamp, alphabetical key values, etc. The multimedia data is stored in a container called 'chunks' or 'descriptors'. A chunk is usually storage of massive data having some similar property collected together. For example, the multimedia data can be categorized into different chunks of descriptors based on the data integration from different data capturing devices such as audio recorders, video recorders and other devices. The captured data needs to be integrated based on time hence the chunks are synchronized with time chunks. The essential data structure like *<descriptor size, Descriptor Name, Header flags, data>* is needed to organize the chunks. The unstructured data can be saved in such chunks of descriptors. This can be unstructured data; for example, data from RfiD can be in <RfiD> chunk, data from video can be in <Moov> chunk. Jpeg image data can be in <JPEG> chunk etc.

◇ 3.2.3 Protection and Security to the Data

Usually, the data is either a personnel information, business related information, security related information or public domain information. The data may need privacy, protection or security against the unlawful data handling. For example, medical examination is private information of a person, it needs privacy. Another example can be emails written are private information of individuals. The business data needs protection regarding data access. Security most of the times may cover both privacy and protection. The security may also cover the matters regarding storage technologies used, location, coding/decoding of data, data structures, access permissions, access to the history, etc. The log-based data mining can help to understand the access patterns to investigate threats to the data privacy, protection and security.

Every data chuck can have protection feature regarding whom it is accessible to. The Header flags can have Protection and Security bit. In case, the Protection and Security bit are enabled, then respective data chuck can have security descriptor added in it. Different types of security mechanisms can be provided based on the requirements and sensitivity of the data.

◇ 3.2.4 Data Storage Technologies

Different storage technology solutions are available in the technology market; Hadoop

Distributed File System (HDFS) is one of the well-known solutions used worldwide to store Big Data. Typically, the Oracle enterprise Big Data solution includes a mix of open source and Big Data software such as Cloudera with Apache Hadoop-CDH4 with Admin manager, Oracle Manager, statistical package R, Oracle NoSQL community edition and Oracle Enterprise Linux Operating System with Oracle Java VM.

The Big Data being big in size, it is required to store it in multiple databases, data cubes and files in the back-end and integrated front-end. Typical open source Big Data tool like MongoDB can be used to store data with variety from numbers, strings, images and videos. It can be interfaced with programming technologies like Python (pymongo), Java in addition to the commercial technologies.

Data-mining technologies and Big Data integration can generate challenging exploitation of Big Data for numerous applications. The subject-oriented, time-variant storage of data, also called data warehouse, can have number of investigative applications in storage organization and energy related performance analysis tools using data-mining. The OLAP (On Line Analytical Processing) is another technology that can be integrated in the middle layer of three-tier architecture of data warehousing. The OLAP can be relational OLAP or a multi-dimensional OLAP or a Hybrid OLAP. The Big Data integration with data mining can be used in a multi-dimensional model where the designs are integrated with data warehouses and data marts. At the heart of these technologies are the data cuber which are the collection of very large set of facts or measures with number of dimensions having entries or perspective defined by the company to be used for storing the records. The OLAP-based data-mining, also called OLAM, is a technique of interactive and exploratory data mining.

◇ 3.3 DATA MINING WITH BIG DATA

Data mining in Big Data can be termed as, looking for something very small, in something which is very big in terms of size, variety and rate of data gathering. For example, it can be compared like gathering knowledge of earth like planets in the Universe, in astronomical science. Universe is a Big Data and earth is very tiny information having certain characteristics of the Habitat. The Universe consists of dust, vapours, asteroids, stars, planets, comets, black holes, supernovas and other unknown maters contributing to the variety. This information data is in multi-trilians of records. This is unstructured data. Another example is to construct an e-newspapers or a blog. An e-newspaper or a blog may demand images, video clips, audio clips to register the opinion or comments, animations, texts in different backgrounds, colours and fonts based on the emotions and context associated with the content, etc. This is unstructured data. The selection of context related information related to the main content requires mining rather than searching to get the effectiveness of a blog or an e-newspaper.

◇ 3.3.1 Data Mining using Pattern Analysis with Big Data

Let us take example of recommendation based on a blog or friend on Facebook. We get recommendation that the visitors visited this article of the blog have also visited another '*xyz*' article on the blog or persons visited this page of Facebook have also visited some other '*abc*'

page of Facebook. This is an example of mining the access pattern. The data-mining can be done with the help of frequent patterns resulting in discovery of association and correlations among the items in the datasets. Consumer shoppe can be the best example for pattern analysis. For example, a person who purchases, say, an HDTV also purchases a dish for different TV channels. This establishes an association with the item sets HDTV and dish channels and discovery of such associations may lead to better marketing strategy leading to more profits to the business.

Association rule: {HDTV → dish channels | *support* = 10%, *confidence* = 60%} (3.1)

The associative rule discovers that most of the customers who have purchased an HDTV have also purchased the dish channels and their percentage is 60 per cent out of total such purchases; also out of total transaction of purchases, 10 per cent of transactions support such associated purchase.

The Big Data interface gives very clear picture to the customers for the choice by presenting the product in virtual view; for example, images of HDTV, 360 degree views of the product, body colour, detail specifications of the product, comparative analysis of the products in terms of selected specification such as costs, screen resolution of the HDTV, etc. The heterogeneous data such as images, animation, strings, text and numbers in addition to multimedia can be stored using Big Data and used as parameters of investigation by composing run-time query for such heterogeneous specifications. Parameter-based data composition using Big Data will be done by the customers resulting into favourable transaction of sale of the goods, and data mining discovery identifies the association of items in the datasets used by the customers resulting into the support and confidence over the association of the products giving valuable support in designing the business strategies.

Let $C = \{c_1, c_2, ..., c_n\}$ be the set of body colours of the HDTV (H); hence the color can be selected in nC_1 ways. Let V be the set of 360 views of the HDTV (H) and let $D = \{d_0, d_1, ..., d_m\}$ be the schemes of dish channels. The customer has a choice of colour with satisfaction of 360 degree views of the product to be purchased. Now, the support and the confidence can be obtained to discover the choice of colour for a given HDTV (H) with conditional probability $P(H/c_i)$. Also, since $H \rightarrow D$, we have,

$$\begin{aligned} \text{Support (H} \rightarrow \text{D)} &= P(H \cup D) \\ \text{Confidence } (H \rightarrow D) &= P(D/H) \end{aligned} \tag{3.2}$$

Hence, the discovery of customer's choice with body colour of the HDTV and the dish channel schemes can be done helping in the design of the business growth plan, ordering strategy and combo-offer as shown in Table 3.1.

In Table 3.1, those who have selected HDTV and opted for channel also selected the schemes (d_i). The first row of the table, 4 per cent of support indicates those who have purchased HDTV and channel schemes have also purchased the dish channel schemes. Also, 40 per cent of total transaction selected the dish channel schemes.

The Big Data mining takes it ahead; to illustrate it further, the study of how the decision is taken either by using advertisement from newspapers, advertisement from TV channels, advertisement from Internet etc., and related frames in the video or Internet video or sound frames in the radio. Each multimedia stream has different data-structure requirements

Table 3.1 Knowledge Discovery through Big Data Mining

Item DataSet Association		*Knowledge discovery*	
HDTV (H/C$_i$)	***Dish Channels (D)***	***Support***	***Confidence***
H/C_0	d_0	4%	40%
H/C_0	d_1	6%	70%
H/C_0	:	:	:
H/C_0	d_m	3%	53%
H/C_1	d_0	2%	55%
H/C_1	d_1	5%	71%
H/C_1	:	:	:
H/C_1	d_m	3%	45%
:	:	:	:
H/C_n	d_0	4%	42%
H/C_n	d_1	8%	79%
H/C_n	:	:	:
H/C_n	d_m	3%	51%

and different complexity requirements. Conventional databases do not support such schemes, but Big Data is all about such integration and support.

Big Data Contributions in Developing Confidence and Support

Let us extend example of HDTV purchase, but now using online application. The 360 degree view and virtual provision of channel selection to run virtual movie or TV program trailer facilities can be provided using Big Data, and the selection HIT count or visit count can be one of the sources of generating confidence and support. Since such application is available on Internet, it can be accessed through different gadgets connected to internet such as mobile devices, desktops, etc. and very large HIT count may result in very large volume of data in very short span, i.e., with very high velocity with varying patterns generated due to available combinations. Analysis of such patterns helps the business company or seller to generate the confidence and support for implementing different business schemes.

Blending the data variations including supported data types and streams is one of the important characteristics of Big Data. In the HDTV example, the specifications of HDTV and channels are text data. 360 degree view and related virtualization are multimedia data and based on the HIT count of the different combinations used by the user, generates the numerical data, related to the confidence and the support. The flexibility of the Big Data to support variety, volume and velocity makes it one of the important tools in knowledge generation. Data mining of the information to generate the knowledge of current patterns on demand and producing the patterns on demand can earn success to a business. For example, the body colour of the HDTV selected by the users can give the data mining pattern about the colour selection trend

in different genders of different age groups, it may vary based on the region and other such parameters. The Big Data has the ability to construct support with the help of flexible data descriptors integrating the multi-streams, multi-dimensional data.

Apriori algorithms are conventionally useful for identification of frequent patterns using Boolean association rules proposed by R. Agarwal and Shrikant in 1994. It works on a prior knowledge. Let $I = \{i_0, i_1, i_2, \dots i_n\}$ be the set of n items for sale in a shop such that i_1 = HDTV and i_2 be the dish antenna.

Schemes D as discussed in Equation 3.2 and Table 3.1. Let A be the transaction matrix showing object association count by $a_0, a_1, a_2, a_3, a_4, \dots, a_n$ as shown in Table 3.2. Let A: $a_i \rightarrow a_i$ be a subset of transaction set A having mapping on itself then it is indicated by 1 else the transaction is indicated by the count a_j. The support and the confidence can be calculated using Equations (3.1) and (3.2).

Table 3.2 Transactions Matrix along with Association among Objects

Objects	i_0	i_1	i_2	...	i_n
j_0	*1*	a_1	a_2	a_3	a_4
j_1	a_5	1	a_7	a_8	a_9
j_2	a_{10}	a_{11}	1	a_{13}	a_{14}
:	:	:	:	:	:
j_n	a_{n-5}	a_{n-4}	a_{n-3}	a_{n-2}	1

The itemset I can be formed as shown in Table 3.2 and tested for the support and the confidence and single item purchase such that $\forall i \in I$ where $a_i = 1$ indicater support of the single item purchase whereas when $a_i \neq 1$ indicates the support for those customers who have purchased ith object and purchased the other item to formulate the itemsets. The Apriori assumes the principle that the non-empty itemsets of the frequent items or items having high HIT ratio with respect to the total purchase, are also frequent. This results in knowledge discovery of patterns adopted by the customers and associated resources used by the customers to conclude the item-set formation as described in the following equation.

$$f(x_{ij}) = \begin{cases} a_i = 0; \text{Significant support} \\ a_i = 1; \text{Individual support} \end{cases} \tag{3.3}$$

The Big Data opportunities can contribute in subdividing the database further in various ways such as survey, screen button click events, contact calls, etc. For example, the consumer trials on the screen to know the combinations of HDTV and dish channel selection. The hit ratios of various combinations of different features are available for the combination of the most preferred product for the consumer.

Multi-level Apriori Big Data processing

Multi-level Apriori algorithms are useful methods for more accurate knowledge discovery in Big Data. One of the characteristics of Big Data is that its size is in peta-bytes. This results in a challenge for knowledge discovery. Multi-level apriori processing divides the

knowledge discovery into multi-level to reach the desired knowledge. The intermediate findings of knowledge can be shared across the levels.

Nowadays, Big Data are created for knowledge discovery. For example, in an electronic Shoppe, 80 per cent of customers who purchase computer may also purchase printer; it could be informative that 71 per cent of customers who purchased laser printers if they bought 12 per cent of Computer. This knowledge discovery requires multi-level data mining for the knowledge discovery. Suppose this shopping mall transaction database has two relations;

- Description of sale items consisting the data structure with set of attributes; bar-code, category, brand, prize;
- Sale Transaction ID and the set of per cent items sold along with the quantity.

The process of mining association rules is designed for discovery of large patterns and strong association rules at the top most concept level for the used Big Data. If the minimum support is set to 5 per cent and minimum confidence is set to 50 per cent, then very large database table is expected for the second level for the item category = HDTV. But if for the first level, the HDTV is searched and at second level sold items are 12 per cent and at the third level, the laser printer then this multi-level data mining on the Big Data results in reductions in the data records.

Association rules from frequent patterns

Equation 3.2 gives the support and the confidence calculation for the items under consideration from frequent itemset. Equation 3.3 presents the transaction ID and support gained by the purchases done by the customers. Hence, $\forall i \in I \sum a_i$ gives the associative support per transaction ID. The $\sum a_i/\text{Count}(i)$ results in the confidence that customers purchased certain item say $b1$ have also purchased the item $b2$ from the item set I. Hence, associative rules can be formed from frequent patterns for better results.

The association rules reduce significantly the item-data set size resulting in better performance. This method suffers due to two reasons. First, after reduction of the item-data set size, still the size can be significantly large because larger the size better the accuracy. Second, it requires repeated scan of the whole item-dataset per pattern. This results in a worst-case time complexity due to large number of comparisons and loops. To avoid worst-case time complexity, a divide-and-conquer strategy is used with the help of tree structure implementation. The tree structures sub-divide the item-data set using support key resulting in either left-child node or right-child nodes, hence improving the search time complexity. Such use of tree structure for pattern growth is also called seed growing or pattern growing method for mining frequent patterns.

This Big Data is all about voluminous data which is collected at large velocity and has variant data structures, it becomes further tedious to handle it using trees. The Big Data uses divide-and-conquer techniques and sub-divides the data into meaningful data chunks or descriptors. The chunk or the descriptor tables are used based on the characteristics and such tables are accessed with the help of the descriptor key called descriptor name. This is explained in Section 3.1.2. Since index-key is used for selecting the table, the redundant searches and comparisons can be avoided.

◇ 3.3.2 Data Mining using Classification Analysis with Big Data

Data mining using classification analysis is a data-mining method which involves data analysis that extracts models describing the important classes or classifiers. For example, in an exciting event of a one-day cricket match tournament between two teams, the last over can be given to a bowler whose performance in last-over against the team or the batsman is excellent. For better results for the confidence, large amount of dataset regarding support is needed. The Big Data has the ability to store voluminous data, hence suitable for such transactions. Also, in Big Data, data descriptor tables are used hence large data can be sub-divided into large number of small tables resulting into meaningful information. The support and confidence can be used as names of the descriptors, hence resulting in better performance. The classification and selection of the descriptor table can be done in various ways using conventional data mining techniques used for classifications such as;

- Decision tree induction
- Bayes classification methods
- Rule-based classification methods

Model-based methods of evaluation and selection of descriptors.

◇ 3.3.3 Data Mining using Cluster Analysis with Big Data

Data mining using cluster analysis is a method which involves data analysis that extracts the data itemsets into meaningful partitions. The set of clusters resulting due to such partitioning are called data clusters. Clustering has a challenge of efficient dataset formation. Based on dataset formation, there are different names to the clustering. Since the cluster is a collection of similar data object set which is dissimilar to the other cluster, it can be called as automatic classification clustering. Based on the similarity, the data objects can be partitioned in the memory, hence the clustering can also be called as data segmentation method. The statistical methods are focused on distance-based clustering methods. The clustering is known as unsupervised learning, hence it is learning by observation.

The cluster analysis can be done using following methods:

- Partitioning Methods
- Hierarchical Methods
- Density-based Clustering Methods
- Grid-based Clustering Methods
- Probability-based Clustering Methods
- Dimensionality-based Clustering Methods
- Graph and Network-based Data Clustering Methods

◇ SUMMARY

The Big Data has three characteristic features, viz. volume, velocity and variety of data. The data is sub-divided using divide-and-conquer to convert it into meaningful data and stored

in the data descriptor tables. The descriptor name keys are generated and used for accessing the descriptor data. The data-mining can exploit this feature of Big Data by organizing the data by support and confidence in the descriptor tables. Since these tables are specific to the support and confidence, relatively, they are smaller in size than sequential tables. Hence, it is effective in space complexity. Also running a search query on very large sequential tables adversely affects the time complexity due to number of comparisons and iterations required. In Big Data, since the data is sub-divided and accessed using descriptor ID or name, it results in better time complexity by reducing redundant comparisons and loops. Also, descriptors are configurable based on behaviour or context, and multiple instances can be created. Hence, Big Data is very useful for data mining.

Multiple Choice Questions

1. Following parameters are mainly used for defining the Big Data.
   (a) Size, data structures, functions
   (b) Volume, velocity, variance
   (c) Concurrency, voltage, volume
2. Data mining usually uses following two conceptual terms for knowledge discovery.
   (a) Confidence and support
   (b) Searching and sorting
   (c) Time and space complexity
3. Following is a classification method used in data mining.
   (a) Decision tree induction
   (b) Searching
   (c) Overlays
4. Following is a clustering based mining.
   (a) Graph and network-based data clustering
   (b) Function clustering
   (c) None of these

Concept Review Questions

1. Create a Big Data for a food mall that sell milk and bread of different brands. Develop a data mining operation to discover the knowledge for identifying the sales pattern for the customers who purchased milk and bread.
2. Create a Big Data for a food mall that sell milk and bread of different brands. Develop a data mining operation to discover the knowledge for identifying the sales pattern for the customers who purchased milk and wheat bread using multi-level data mining.

4

Long Live the King of Big Data
The Context

—DR. ANAGHA KULKARNI

◇ 4.1 INTRODUCTION

During elections, news channels scientifically analyze election trends. Each channel wants to be the first among all to accurately predict the trends. Many people are actively involved in posting their opinions on social media. Therefore, the news channels have to analyze the data from multiple sources such as opinion polls, surveys, Twitter, Facebook, WhatsApp, blogs, on line advertisements, reports, audio interviews, news articles, etc. In this scenario, data comes in different formats such as questionnaire, tweets, messages, images, unstructured text, audio files, etc. Data is bombarded. Moreover, data could be conflicting and noisy. It is challenging to process relevant data and come up with accurate predictions in least time. Edna Ferber, an American novelist, rightly says, 'Perhaps too much of everything is as bad as too little.'

Ninety per cent of the data today has been generated in last two years. Not only this, 80 per cent of this data is unstructured. This is a result of increased use of mobile devices. Number of smart phones and tablets have exceeded number of laptops and personal computers. With improved and affordable internet connectivity and ease of use of apps such as Facebook, Twitter, YouTube and so on novice users are equally active in generating data. As a result, data is turning out to be Big Data.

Human beings are the best pattern recognition machines. Naturally, they tend to create patterns. When tweets, blogs, posts, articles, news and messages are written, they are bound to have patterns. With the increasing amount of unstructured text, it is very important to discover patterns in the text. One can find patterns in usage of words, smileys, hash tags, etc.

Data that is generated by Facebook posts, tweets, emails, blogs, ratings, reviews, reports, etc. is unstructured. Such data does not have any specific format and does not fit into any pre-defined schema. Tremendous increase in the volume (from KB to YB), variety in the formats

of the data (from structured to unstructured) and velocity with which the volume is increasing (from batch to real-time) make it difficult to understand and make sense out of it. Data having all the above characteristics is called Big Data. Figure 4.1 shows three Vs of Big Data.

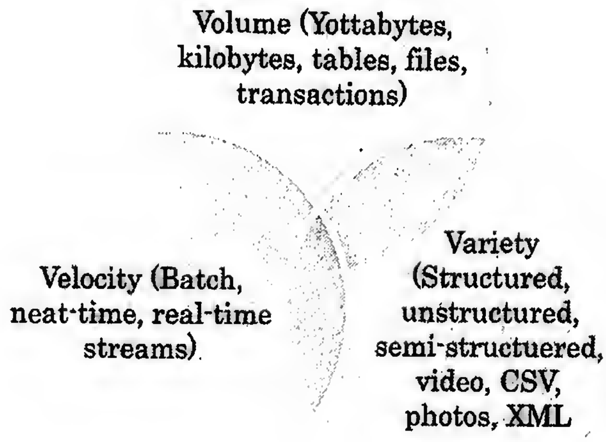


Figure 4.1 Three Vs of Big Data.

The big question is, 'Is the data that being analyzed or used meaningful, accurate and sensible to the user?' It can be made sensible, relevant and useful only if it is **Context**-based rather than only **Content**-based. Figure 4.2 shows relationship among content, context and relevant information.

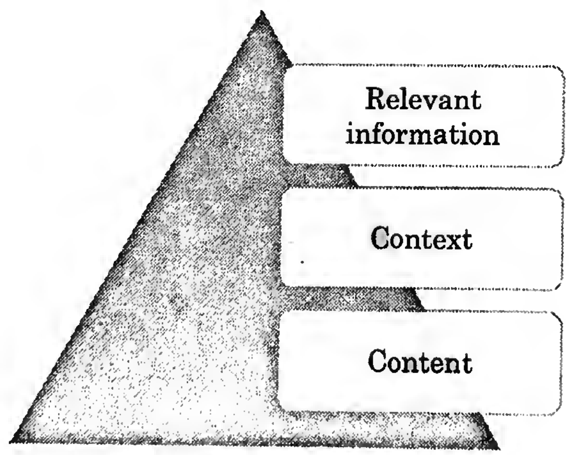


Figure 4.2 Relationship among content, context and relevant information.

From this discussion, it is clear that Big Data is humongous in size and relational databases cannot scale to such a huge volume. Moreover, relational databases cannot handle such a huge variety of data with required amount of speed. Hadoop and MapReduce are suitable for handling all the three Vs of Big Data. This chapter presents a case study on context-aware recommender system using Hadoop.

This chapter is organized as follows. Next section states definitions of context as stated by multiple researchers. Section 4.3 emphasizes the importance of context in Big Data. Section 4.4 states how to use contextually enabled data. Issues in use of context are specified

in Section 4.5. Context types are presented in Section 4.6. How to find context in user data is discussed in Section 4.7. Section 4.8 discusses the methods to discover closeness in large and short text. Section 4.9 reviews contextual analytics. Section 4.10 briefly touches upon privacy and security of Big Data. Sections 4.11 and 4.12 present case studies. Section 4.13 concludes the chapter.

◇ 4.2 WHAT IS CONTEXT?

Context is defined by many researchers depending on the application in which it is used. Some of the definitions are as follows.

- Context is conceptual garbage can.
- Context is any information that can be used to characterize the situation of an entity. An entity is a person, place or object that is considered relevant to the interaction between a user and an application, including the user and applications themselves.
- Context is referred to as location, identities of nearby people and objects and changes to those objects.
- Context is defined as location, environment, identity of people and time.
- Context is the set of environmental states and rules that either determine an application's behaviour or describe where the event occurs.
- Context is defined as an application environment or situation.
- Context is the history of all that occurred over a period of time and small set of things they are attending to at that particular moment.

In summary, context is nothing but user's preferences which are infinite, but only partially known. It is situational information. It is not clearly specified in the data. It has to be derived without user interaction from relationships between data and other situations such as source of data, creator of the data, time of creation, place of creation and recipients of the data, etc.

◇ 4.3 IS CONTEXT IMPORTANT IN UNSTRUCTURED BIG DATA?

User generated data has rich metadata, describing user's personal interests, preferences and friendship relationships and certain patterns. User's sphere of interest and activities, preferences and friendship relationships can be analyzed without user interaction. Thus, we can say that context is inherent and deep rooted in Big Data. Data generation occurs in specific environment, at specific time, by a specific user, under specific weather conditions. So, if this humongous amount of situational information of the data is used, it is very useful in analysis. Data becomes meaningful, relevant, accurate and sensible to the user. Chris Anderson, a British-American writer, rightly says, 'In a world of infinite choices, context—not content—is the king'. Figure 4.3 shows different types of data which can be used as context of Big Data.

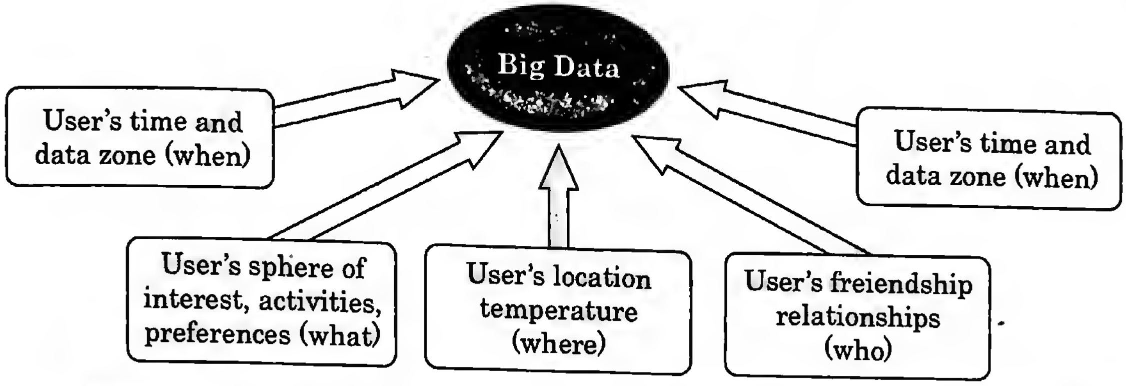


Figure 4.3 Different types of data that can be used as contextual information.

◇ 4.4 HOW TO USE CONTEXTUALLY ENABLED DATA?

When data is teamed up with context, many things are done for you as opposed to done by you, making user's life easier. Sensors and apps play the role of silent observer. They observe what you do and how you do and provide relevant information to you at right time. Making use of the situations, recommender systems and advertisements can be useful to give suggestions to the user. For example, a user is interested in listening to classical or jazz music, based on the surroundings and his mood, a recommender systems can recommend the music he is most likely to listen. In this case, user does not have to search for music. It is equally true with advertisements, say a user is roaming around in Hawaii and it is almost lunch time, context-sensitive advertisement can suggest him some restaurants in nearby area using location, time related contextual information along with his preferences.

◇ 4.5 WHY IS CONTEXT A BIG ISSUE IN UNSTRUCTURED BIG DATA?

The most obvious question is if context is inherent and deep-rooted in Big Data and if it so useful, what the talk is about? Why is it a big issue? Answers to these questions are not so easy though. It is clear that value of data increases if it is augmented with context. By taking into account the context, data becomes self-explanatory and new insights can be discovered. With worthiness, come the difficulties in deriving the context. Following are the issues that are faced in deriving the context:

1. Managing and organizing context efficiently is a big challenge. Contextual data increases so rapidly that managing history becomes difficult. Another question is how long should a piece of information be considered useful and preserved?
2. Defining relevancy of the contextual factors in present situation from massive amount of raw data.
3. Selecting relevant information has to be done intelligently. A particular piece of information may be useful and relevant only for short amount of time at other times, it may be considered as noise. This indicates user interests could be short-lived or long-lived. As an example, consider that during a tournament a user might be interested in

getting updates about current scores and physical conditions of his favourite players. Once the tournament is over, usefulness of that information decreases. Recency of events plays an important role in this situation.

4. Big Data contains large text documents as well as short texts from tweets, SMSes, etc. Even though large text documents generally have standard writing style and standard spellings, due to uncertainty in the size, they are difficult to handle. However, lot of work has been done in this area already.
5. Short text contains very limited characters (up to 140). It is very noisy. A study by Pear Analytics concluded that 40.55 per cent of tweets are pointless babble and 5.85 per cent are self-promotion messages. So, they may not have any topic. They are fast changing. Following are the limitations of short text messages:
   (a) There is no standard format, and writing style is informal. They may neither have any paragraphs and sentences nor any punctuation marks and case-sensitive text. This makes it very difficult to understand if user is speaking about a place, another user or thing.
   (b) Abundant use of special characters such as smileys, #, @, etc. Sometimes URLs are also found in the texts.
   (c) There is no notion of single standard spelling. Variety of spellings differs when spelled in American and British ways. In addition to that, young generation has created a new texting language. It was observed that 'tomorrow' was spelt in 16 different ways by users in a corpus of 1000 SMSes. Table 4.1 shows different spellings of 'tomorrow'.

Table 4.1 Different Spellings of 'Tomorrow' in a Corpus of 1000 SMSes

Sr. No.	*Spelling of tomorrow*	*Frequency of occurrence*
1	Tomoz	25
2	Tomorrow	24
3	Tomoro	12
4	2moro	9
5	Tomrw	5
6	Tomora	4
7	Tomo	3
8	2maro	3
9	2mro	2
10	Tom	2
11	Tomra	2
12	Tomor	2
13	Tomm	1
14	Morrow	1
15	Tmorro	1
16	Moro	1

There are many such words having different spellings such as b4, w8, u r, I m, thx and so on.

(d) Slangs are widely used. Table 4.2 states a few slangs.

Table 4.2 List of Slangs

Sr. No.	*Slang*	*Meaning*
1	Aap	Always a pleasure
2	Aip	Am I pretty
3	Btw	By the way
4	Dp	Display picture
5	Dway	Dude who are you
6	Fyi	For your information
7	Iddi	I didn't do it
8	Lmgo	Laughing my guts out
9	Lol	Laugh out loud
10	Nmf	Not my fault
11	Rip	Rest in peace
12	Tc	Take care
13	w/o	Without
14	Way	Who asked you

(e) Regional dialects have influence on short texts. Multi-lingual texts are written. New words are introduced due to cultural influence; there could be mixing of languages, corruption of words and sentences.

All these factors make it difficult to decide usefulness and relevancy of piece of contextual information for the user.

◇ 4.6 CONTEXT TYPES

Context can be classified in different ways. The definitions and concepts are overlapping.

(a) **Location context:** Location context is the information about a person's whereabouts. This piece of information is very important. Once this is known, many other pieces from the puzzle fit properly into their slots. For example, if it is found that a user is in Australia and if he is sending a message to another person in America, it is important to inform the user that he may not get immediate reply from the other person. Location context may also be able to discover whether user is moving or stationary.

(b) **People context:** People context is the information that finds which other people the user is in touch with.

(c) **Object context:** Object context is software components that the user uses. Software components such as files and applications give more information about the user's likes and dislikes. For example, on which applications user spends most of his time and which files (books, audio, video, etc.) he accesses most.

(d) **Social context:** Social context includes reactions of other people or applications that are in direct or indirect contact with the user. This may also include the way user prefers to communicate with other persons or an application, who are close friends of the user and on which application. This may also include different date and time formats. Social context also includes cultural context which may include usage of words, phrases and slangs.

(e) **Spatial context:** Spatial context refers to the environmental conditions of the user. This includes location, temperature, noise, light effects and so on. This may help to find where user currently is.

(f) **Temporal context:** Temporal context refers to the time when tasks are carried out. If the schedules and deadlines are approaching, the user can be alarmed. A judgment can be done whether user accesses an application periodically or rarely.

(g) **Mobile context:** Smart devices have sensors and apps. Whether you want it or not, sensors produce lot of situational information. Similarly, social networks produce lot of unstructured data. We get sensor data from the following:

- Satellite images (such as Google Earth)
- Scientific data (such as location, weather)
- Photographs and videos (surveillance, traffic video)
- Radar and sonar data.

Sensor data helps to understand whereabouts of the user, based on which recommendation of services can be done. Sensor data gives weather, location, time of day, user mobility (moving or resting), heart rate and blood pressure. Use of keys/touch screen may indicate mood/mental condition of user, etc.

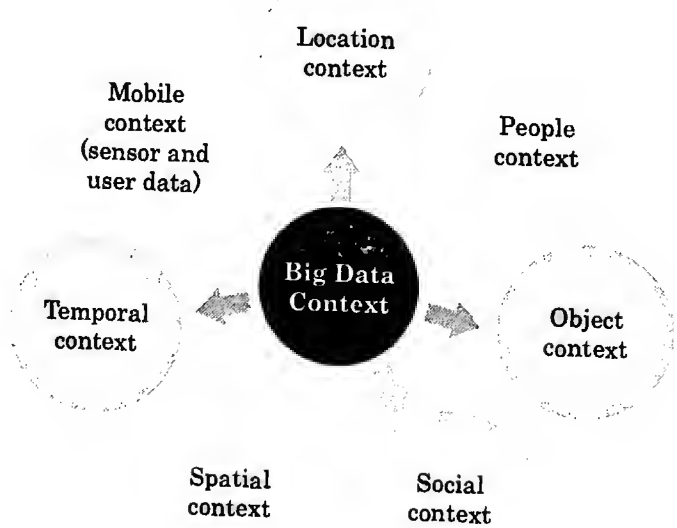


Figure 4.4 Different contexts in Big Data.

Similarly, user generated data is due to:

- Social networks like Facebook, LinkedIn, MySpace, Flicker, etc.
- SMS, emails, survey reports, review and product information.

Ratings help to understand the likes, dislikes and preferences. Facebook and LinkedIn friends indicate friendship choices of the user. Google calendar helps to understand user occupancy and free time. Using sensor and user generated data, user context can be built. Figure 4.4 summaries important contexts in Big Data.

◇ 4.7 CONTEXT IN USER DATA

Big Data contains large documents such as reports, reviews, and news articles and so on. Such documents have sentences and paragraphs. They have punctuation marks and use standardized spellings and way of writing. Big Data also contains short text data from tweets, Facebook posts and so on. Such data does not follow any standard way of writing and most importantly they have very few characters. Due to the large difference in their sizes and writing style, techniques for finding context are different. They are discussed below.

◇ 4.7.1 Identification of Context Region in Large Texts

When a document is written, it is divided into three logical parts—introduction, details and conclusion. These sections have distinct purposes in defining intentions of documents. It is a well-known fact that an introduction or lead paragraph defines the topic and establishes context. Subsequent paragraphs contain detailed information. Concluding paragraphs close the topic. Therefore, it is very important to understand in which sections the term occurs.

Context of large unstructured documents can be found in two ways:

Intra-document information: Using information about the words in the document

Information about word position or even its formatting can also be used to find context. Word can be considered contextually important if it appears in title or at the beginning of the document. Also, proper nouns can be considered important. However, this method supposes that only several sentences, which are located at the front or the rear of a document, have the important meaning.

Context can also be found using important sentences in the document. Importance of a sentence is measured by two methods. In the first method, the similarity between sentence and the title is found. In the second method, importance of a sentence is found by using importance of words using TF, IDF and chi-square.

Contextually important word can be found using its position in sentence, position of sentence in paragraph and position of paragraph in text. First position in sentence, paragraph and text can be given highest weight. Thus, first word of the document is considered to be the most important word contextually.

Depending on whether a word is compact or distributed, its contextual importance can be decided. An important word is more likely to be spread over the document than being compact, so when a word is distributed, it is given more importance.

Another way is to create Contextual Positional Regions (CPR) in the document. Contextual Positional Influence (CPI) of a word can be decided depending on the CPR of its occurrence. CPRs can be created using CPR size or discourse segmentation.

Another consideration is where a word appears first in the document. It is based on the common premise that important contents are mentioned earlier in the document. Thus, more importance is given to a word if it appears earlier.

Inter-document information: Using information about the document

Unstructured text documents span from a few sentences to few pages. In such cases, the most common ways of finding context is to find the author of the documents, language, readability index, genre and so on. Many times, user's sphere of interest can also be decided using his other activities on smart phones, tablets and other devices. Other activities such as which other documents the user reads, how has he organized all the documents within his device, etc. Using such information, it is easy to find sibling, child and parent folders and documents within them to decide user's sphere of interest. Both the methods make use of environmental information.

◇ 4.7.2 Identification of Context Region in Short Texts

Techniques discussed above cannot be applied to short texts since they rely on word position in the document. Short texts have 140 characters. There is no notion of grammar, punctuation marks and spellings. So, new techniques are required to find context in short texts. Short texts have special characters such as smiley, @, # and URLs. These are very helpful in finding the context of the texts.

(a) **Smiley:** Smiley, also known as emoticon, indicates the mood of the author. Using smiley is a new way of expressing emotions.

(b) **#:** Before any relevant keyword or phrase (without spaces) tweeters use the **hash tag** symbol # (For example, #presidentialelection). It highlights the context in which tweet is written. All other tweets having same context can be found by clicking on a hash tagged word in any message. Hash tag occurs anywhere in the message.

(c) **@:** @ sign is used before usernames in Tweets (For example, @stevejobs). To refer to a person in the tweet @ is used. This sends a message to that person on Twitter. @ can also be used effectively to derive context about the tweet.

(d) URLs are the part of tweets. Tweets having similar URLs can be said to have similar context.

(e) Short texts are influenced by regional dialects. They can predict the location of tweet authors. They reveal regionalism. For example, Southerners' commonly use 'y'all,' whereas Pittsburghers' use 'yinz'. Some people call soda, pop, whereas some call coke, based on area in which they live. In northern California, something that is cool is 'koo' in tweets, while in southern California, it is 'coo'. In many cities, something is 'sumthin', but tweets in New York City favour 'suttin'.

◇ 4.7.3 Closeness

Closeness indicates similarity. Closeness between two large texts or short texts helps us to find similarity between them, and therefore, the context of the documents. As discussed earlier, there are differences in sizes and writing styles of large and short texts. Thus, the techniques of finding closeness differ.

Large texts

Similarity between two documents is most commonly found using distance. Distance functions that are commonly used are Euclidean, Manhattan and Minkowski distance. Similarity is also found using cosine similarity. However, this way of finding similarity between two documents might not be suitable always.

Closeness between two documents can be found using pattern of occurrences of words. Correlations may exist between two documents which are far away from each other distance wise but have similar patterns.

Consider sample patterns as shown in Figure 4.5. In the first case, all the three patterns are close to each other distancewise and are similar. In second case, bottom two patterns are very close to each other and all are similar. In the third case, third pattern is shifted. Even though, the patterns are similar, patterns are also scaled.

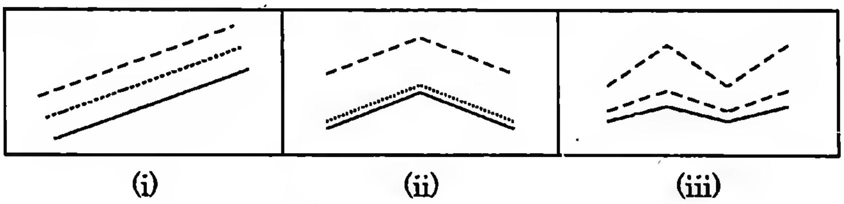


Figure 4.5 Simple patterns.

When distance is used to find closeness between patterns, in first case, all patterns will be found to be similar. In second and third cases, bottom two patterns will be considered similar but third pattern in both cases will not be considered similar. If similarity of patterns is used to find closeness, in second and third cases, all the patterns will be found to be similar.

Many researchers have found patterns by mining frequent termsets (similar to itemsets). Using algorithm for association rule mining, frequent termsets are found.

Closeness factor is another way of finding closeness between two documents. It is a probabilistic technique. It calculates closeness between documents by analysing them. Closeness factor compares patterns of occurrences of words in the documents and calculates closeness between them.

Short texts

To find closeness between two short texts, traditional methods applicable to large texts cannot be used. However, researchers have found different techniques to find closeness between them.

Short texts can be inflated with additional information to appear as if it is normal text document. Traditional techniques can be applied in this case. This method is time consuming and not suited for real time applications.

In another technique, each word from tweet is mapped to Wikipedia pages and their categories. Amount of overlap gives closeness between two tweets. This method is not time consuming. It gives fairly accurate results. The concept is described using Figure 4.6.

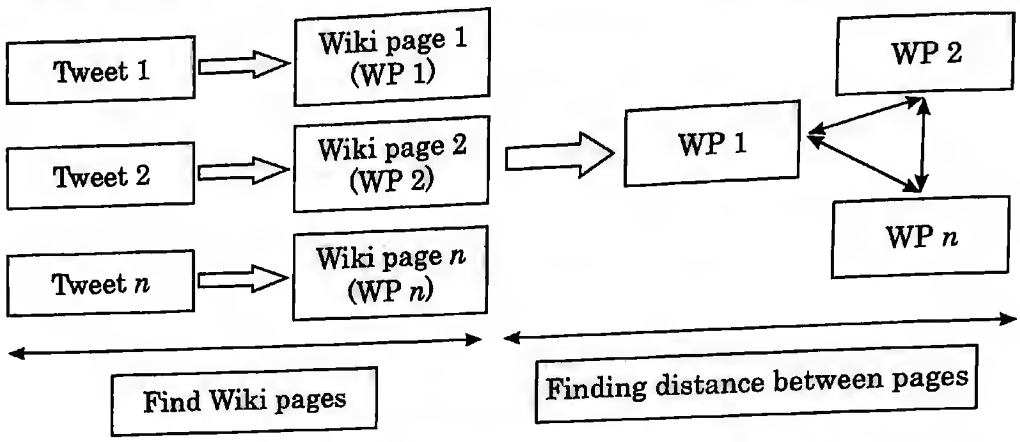


Figure 4.6 Finding closeness between two tweets.

Twevent makes use of tweet segment instead of only single words (For example, steve jobs or happy new year). It then maps every tweet segment to Wikipedia pages and their categories. Closeness is found by amount of overlap between tweets.

Having understood what context is, how important it is and its classification, it is now important to understand how to analyse using context.

◇ 4.8 CONTEXTUAL ANALYTICS

Contextual analytics is an ability to convert data into knowledge. Knowledge is something that we gain by analyzing information and applying context to it. Assume that a traveller is in Hawaii.

Data: Location: 21.3114° N, 157.7964° W, Time: 12 pm (mobile sensor data).
Information: User is in Hawaii and it is noon (derived from data).
Knowledge: Analyzed information–User will look for restaurants (prediction).
Context: He likes pizzas and pastas (derived information from history or input from user).
Recommend: Italian restaurant in vicinity (context-based analysis).

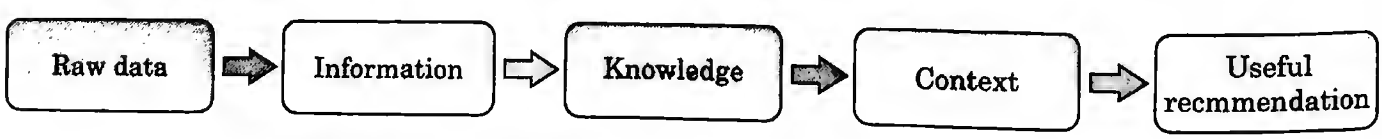


Figure 4.7 Process of contextual analysis.

Figure 4.7 explains the whole process in the form of block diagram. Usability of data increases in this case. It saves users time and helps in making more informed decisions. Contextual analytics is the 'engine behind the Internet of Things'. Contextual analytics adds new relations with previous entities that are related, adds new relations to previously unknown entities and distinguishes between related and unrelated entities clearly.

By using contextual analytics with Big Data, organizations can derive trends, patterns, and relationships from unstructured data and related structured data. These insights can help an organization to make fact-based decisions to anticipate and shape business outcomes.

◊ 4.9 ADVANTAGES OF CONTEXTUAL ANALYTICS

(a) Using context with information delivers higher quality models. As a result, outcomes are useful and relevant.
(b) Real-time contextual analytics helps run time assessments, while the observations are still occurring. Real-time contextual analytics discovers patterns in real time data streams, exploring trends from social media streams.
(c) Using context analytics with Big Data, allows organization to achieve greater success regardless of whether the objective is mitigating risk or recognizing opportunity.

Privacy and security

Not only context strengthens the knowledge, but also it clarifies ambiguity when unstructured data is coupled with context. Historical and latest contextual information can be combined to give a new perspective to data making it relevant. The most important thing is that the all the sensors and the apps need to talk to each other and they have to share the information with each other. Thus, the most important question arises, is this done securely?

Even though the real power of Big Data comes from being able to combine an organization's own data with data outside the company's firewall, according to James Woo, how much data should be shared is the real question. There are lot of concerns on the privacy and security of personal and organizational data. The issue needs to be solved.

Case Study I: Context in Facebook

Facebook is an online social networking service launched in 2004. It is popularly used by 890 million daily active users across the world as of December 2014. Each user spends 21 min per day on Facebook as of 19-Sept-2014. Each user has on an average 338 friends and the median number of friends is 200. If every friend posts at least one comment everyday, on an average each user will get at least 300 plus stories. Facebook says, 'With so many stories, there is a chance that people would miss something they wanted to see if we displayed a continuous, unranked stream of information'. Therefore, the most universal question surfaces is—How does Facebook decide which friends posts' should be ranked first?

To answer the question, Facebook constantly collects data on many aspects.

Personal details

A user's name, email, city, gender, educational institutes, etc.

Usage details

(a) How often a user interacts with friends or public figures?
(b) What is the relationship between a user and his friend (gender, place, educational institute, work place based)?
(c) Which friends in particular a user frequently (or rarely) interacts with?
(d) When a user likes, shares or comments on a post, how much he has interacted with that kind of posts in the past?

(e) The number of likes, shares and comments, a post or image receives from the world and friends.
(f) Whether the object is hidden (every feed has small arrow in top right corner) or reported.
(g) How Facebook is accessed (type of web browser, IP address)?
(h) How long is Facebook accessed and how frequently?
(i) Keywords from a user's stories.

Facebook compiles all this statistics and knows more about his users. This helps Facebook derive contextual information about every user. Using the context, Facebook smartly makes decisions from which friend we would like to know more. It uses algorithm to filter 1500 posts that could be shown in a day. It prioritises 300 news feeds from these. Not only this, it is a feature that banishes people whose updates do not interest you (but you still do not want to unfriend). All this is done keeping in mind what a particular user's context is—what he likes and/or is not interested in.

'Last Actor' looks at the 50 people you most recently interacted on Facebook such as viewing someone's profile or photos, and liking their feed stories. Facebook then shows you more of them in your feed in the short-term. Say you browse through 100 photos of a girl you have a crush on, you will see more of her in your feed later that day. This feature only affects what you see.

'Chronological by Actor' is Facebook's attempt to make real-time content more comprehensible. Say a friend is posting rapid updates about a football game. Showing them in ranked order regardless of their chronological order would be confusing, as you might see the game's final score first, then a photo from half-time, then a touchdown in the third quarter, and then your friend's excitement about the game starting. So, Facebook will soon start to show these rapid real-time updates in chronological order so that you see the first update first and the rest in order.

To show the advertisements that are interesting to a user, Facebook uses all the compiled information. In addition to above information, Facebook finds context of the user, using his/her current location and demographics, which other advertisements do not want to see and such information about the user.

Following are some ways advertisers may build rules on:

- **Keywords:** When a user posts or comments, the keywords that he uses are very important. The keywords from original post are also considered to be important to derive context about the user. For example, when a friend posts about 'nutritious food', the keywords from the original post and the user's post give information about user's likes. It is obvious that the user is interested in staying healthy. So, the advertiser may post advertisements about 'healthy food', 'exercise equipment' and so on.
- **Category information:** User profile has many fields such as sports, music, movies, TV shows, books, etc. where a user can register his likes.
- **Other liked pages:** Other pages that a user likes or is not interested also give contextual information about the user.
- **Places visited:** Other places a user has visited recently also help to gather context.

- **Common interests:** Common interests between a user and his friends' may be helpful in providing contextual information about a user.

Using all such information, advertisements that are more relevant to a user are included in suggested post of the user. In future, it is likely that Facebook will use rich context information obtained from a mobile phone. User's environment (car, restaurant, street, etc.), his activity (idle, running, walking, etc.) will give more information about his mood. This information can be used by Facebook to entertain the user or display relevant advertisement.

Case Study II: Putting Context Aware Recommendation System (CARS) in Practical Use

Predictive analytics in Big Data is the task of extracting information from Big Data and forecasting what may happen in the future. Predictive analytics is useful in organizations to predict the future outcomes and trends. However, challenges in Big Data predictive analytics are:

(a) Ever-increasing data
(b) Constant change in context
(c) Fast response times

To find a solution to these challenges, Intel IT has developed a Context Aware Recommender System (CARS). If predictive analytics is coupled with context, the predictions are more relevant, useful and cost effective. By building the CARS with Intel® Distribution for Apache Hadoop, Intel has shortened time to market, expanded revenue-generation opportunities and built a reusable recommendation engine.

Recommendation systems are a type of information filtering system that try to predict user preferences. The predictions are done based on user actions and history. For example, if a user orders book 'Playing It My Way: My Autobiography' on Amazon.com, immediately new recommendations such as 'The Test of my Life' or 'Rafa: My Story' are flashed. Such recommendations are done using current action of the user and the context derived from it.

◇ 4.10 USING APACHE-HADOOP FOR CONTEXT AWARE RECOMMENDER SYSTEM BY IT@INTEL

Intel has developed CARS using Apache-Hadoop for their Business Units (BU) for recommendations such as advertisements, coupons, reminders, etc. Contextual information that CARS uses is time of day, location, weather, season, device characteristics, etc.

Customers use an application using navigation route or single line search keywords. CARS uses mobile navigation application to provide information to the customers about coupon offerings from restaurants, shops, etc. on his destination route.

Working of CARS

(a) When a customer travels on a route and activates the application, BU gathers coupon offerings on different routes.
(b) Customer-specific preferences are mapped with contextual information.
(c) CARS comes up with the list of most relevant to least relevant point of interest.
(d) The list is presented to the user.

System architecture

CARS uses data mining process involving data pre-processing, data analysis and result interpretation. Data pre-processing and analysis is done offline and result interpretation is done online.

(a) **Data pre-processing:** This is a offline process. Pre-processing involves collecting data from various sources and integration by transformation if required. Pre-processing uses Data Warehouses on shared-nothing and Massively Parallel Architecture.

(b) **Data analysis:** Data analysis is done using parallel and distributed processing across clusters of computers using MapReduce programming paradigm. Algorithms are executed using Apache Mahout on few terabytes of data. This is also done offline.

(c) **Result interpretation:** Final list is prepared online. It includes data retrieval, computational layer and standard application programming interfaces.

Data Flow

Data flow includes three layers, namely, pre-filtering, modelling and post filtering. Figure 4.8 illustrates the data flow in CARS.

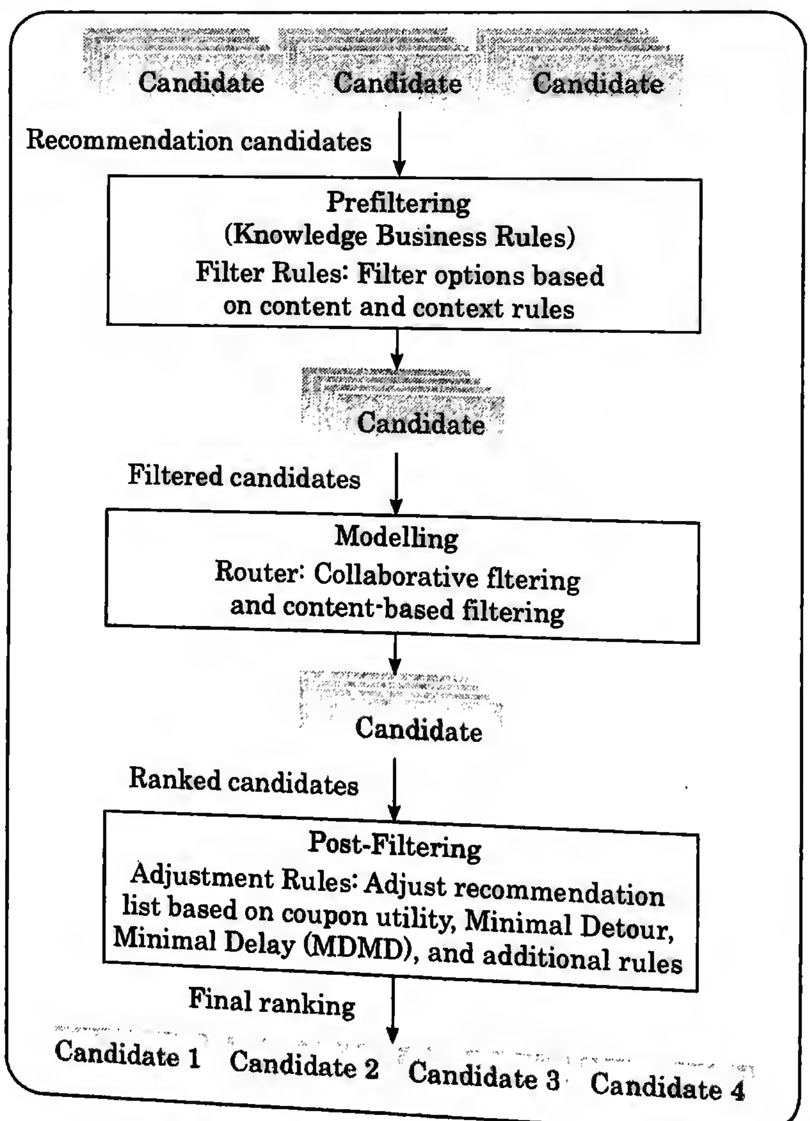


Figure 4.8 Data flow in CARS.

(a) **Pre-filtering:** Before delivering final list, all coupons on the route are processed. Coupons are activated after customer activated the application either through navigation route or SLS. CARS applies filter rules using knowledge based business rules based on contextual information.

(b) **Modelling:** This layer consists of two machine learning algorithms, namely, collaborative filtering and content-based filtering. CARS combines the results of both the algorithms into one model which is used for prediction.

(c) **Post filtering:** CARS applies additional knowledge-based rules to ranked items. This helps in adjusting the score based on current context. This decides which coupons should be presented to the customer.

◇ SUMMARY

Context is inherent and requires in Big Data. Even though use of context in processing Big Data increases value and usability of data, there are challenges in deriving the context. However, lot of research is being made on how to derive the context. In addition to this, huge amount of data is being produced by smart devices of the user. Big giants such as Facebook, Intel, Amazon and the like are already using it to reduce Time To Market (TTM), better decisions and risk management.

Multiple Choice Questions

1. Defining relevancy of the contextual factors in certain situation is challenging because
   (a) Managing and organizing context efficiently are not easy.
   (b) Big Data contains large amount of text.
   (c) Big Data contains large amount of raw data.
   (d) Big Data contains small sized data such as tweets and SMSes.
2. Social context is:
   (a) The application on which user spends most of his time.
   (b) How the user prefers to communicate with others and use of slangs, phrases, etc.
   (c) Environmental conditions of the user.
   (d) Satellite and radar data.
3. User generated data is:
   (a) Data in various apps and emails.
   (b) Data generated by various sensors on mobile devices.
   (c) Data in space.
   (d) Nothing but user's friends.
4. Context of a text document (small or big) can be found using:
   (a) Positional significance of the word.
   (b) Author of the document.

(c) Location of the document.
(d) Smileys and hashtags.

5. To find closeness between short texts:
   (a) Some extra data is added to short texts.
   (b) Only hashtags can be used.
   (c) It is not possible as text is very small.
   (d) Real-time contextual analysis must be done.

Concept Review Questions

1. Explain the relation among content, context and relevant information.
2. What makes context an inevitable part of Big Data?
3. Explain the challenges in using context.
4. What are the different ways to identify context in short texts?
5. Explain different ways to find closeness between large and short texts.

Critical Thinking Questions

1. Explain with the help of diagram five types of data that can be considered as context. Do you think they are relevant to context? Justify your answer.
2. Can you identify the contextual elements in any application (such as music, images, etc.)?

Laboratory Assignments

Input: 100 Tweets/Facebook posts/WhatsApp messages as sample unstructured text messages.

1. Find context of above data. Find the topic of discussion and the person (if any) about whom the tweets are!
2. Write a code to correct slangs. Prepare your own dictionary.
3. Find how many slangs are used in above Tweets/Facebook posts/WhatsApp messages.

5

Big Data: Text Categorization and Topic Modelling

—Dr. Yashodhara Haribhakta

◇ 5.1 INTRODUCTION

With the dramatic growth of text information, such as web pages, news articles, scientific literature, emails, blogs, instant messages, etc., there is an increasing need for powerful text mining systems. These systems would organize the collection of text documents and automatically discover useful knowledge from them. There is an increasing need for going beyond finding text information for discovering novel knowledge from the text data, also known as *text mining*. Typical text mining task includes text categorization, text clustering, concept/ entity extraction, sentiment analysis, document summarization, entity relation Modelling, etc. Text mining and relation modeling is core part of Big Data mining when we are dealing with huge text data.

◇ 5.1.1 Text Mining

Text mining is the analysis of data contained in natural language text. It refers to the process of deriving high-quality information from text. The application of text mining techniques to solve business problems is called ***text analytics***. Text mining can assist an organization in building an accurate business model with deep insights by analyzing the text information, such as text documents, text emails and messages on LinkedIn, Twitter or Facebook (social media). Text data is often said to be ambiguous. The ambiguity can exist due to the inconsistency in syntax and semantics, including text slanginess, languages specific to industry and different age groups, languages with double meaning sentences and sarcasm. Mining of such unstructured data is a challenge for techniques in machine learning, natural language processing or statistical Modelling. Typical text mining task includes text categorization (i.e., document classification),

text clustering, concept/entity extraction, sentiment analysis, document summarization and entity relation Modelling. Text analysis involves information retrieval, lexical analysis to study word frequency distributions, pattern recognition, annotation and information extraction. The goal of these techniques is to turn the text into data for analysis, via application of natural language processing and analytical methods. A typical application is to scan a set of documents written in natural language, and either model the document set for predictive classification purposes or populate a database or search index with the information extracted.

◇ 5.1.2 Text Categorization

Text Categorization (TC) is a discipline responsible for the automatic classification of text documents under predefined categories or classes. TC task lies under the automatic Text Classification problem in Machine Learning. If we use supervised classification techniques, then, there is a predefined set of classes or class, and classification is assumed of training the system on the collection of text documents so that when a new text document is presented to the trained system, it is able to assign the text document to one of the predefined set of classes or class. This technique of supervised classification is commonly known as Text Categorization. There are three paradigms in TC, as shown in Figure 5.1: the binary case, the multi-class case and the multi-label case.

- In the binary case, a text document belongs to exactly one of the two given classes. Thus, the classifier has to determine to which of the two classes the document belongs.
- In the multi-class case, a text document belongs to just one class of a set of m number of classes.
- Finally, in the multi-label case, a text document may belong to several classes at the same time, i.e., classes may overlap through document.

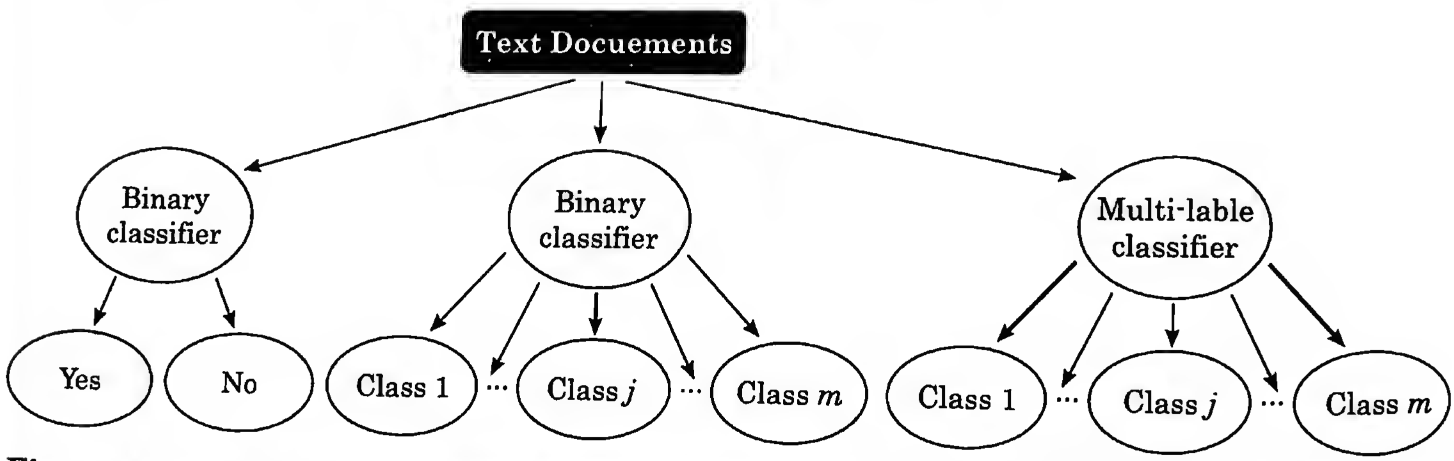


Figure 5.1 Paradigms in text categorization: binary, multi-class and multi-label.

In binary classification, a classifier is trained by means of a supervised algorithm, which assigns a document to one of the two possible sets. These two sets are referred to as sets containing 'belonging' documents, called positive samples, and other containing 'not belonging' documents, called negative samples. The binary case has been set as a base case from which two other cases can be built. In multi-class and multi-label assignments, the traditional approach consists of training a binary classifier for every class and then whenever the binary base case

returns a measure of confidence on the classification, assigning either the top ranked one (multi-class assignment) or given a number of top ranked ones (multi-label assignment).

Formally, text categorization is the task of assigning a Boolean value to each pair $(d_j, c_i) \in D \times C$, where D is a domain of documents and $(c_1, c_2, ..., c_{i|C|})$, and is a set of predefined categories. A value of T assigned to (d_j, c_i) indicates a decision to document file d_j under category c_i, while a value of F indicates a decision not to file d_j under c_i. More formally, the task is to approximate the unknown target function, Φ': $D \times C \rightarrow \{T, F\}$ (that describes how documents ought to be classified) by means of a function Φ: $D \times C \rightarrow \{T, F\}$, called the classifier (aka rule, or hypothesis, or model) such that Φ' and Φ 'coincide as much as possible.'

It has enormous real world application. For example, news articles are usually categorized based on news topics or **geographical codes;** research papers are often classified by **technical domains** and **sub-domains**; healthcare organization categorizes patient reports based on disease categories or types of surgery or insurance type and so on. Another known application of text categorization is spam filtering, where email messages are classified as **spam** and **non-spam**, respectively.

In general, the automatic text categorization can be defined as assigning pre-defined category labels to new documents based on the likelihood suggested by a training set of labeled documents. It generates a classifier from the training set based on the characteristics of the documents already classified. Then it uses the classifier to classify the new documents. Using this approach, we can categorize the documents. Real-world applications of text categorization often require a system to deal with tens of thousands of categories defined over a large taxonomy. Since building these text classifiers by hand is time consuming and costly, automated text categorization has gained importance over the years.

◇ 5.1.3 Context Learning

Text is usually associated with rich context information. Context is useful in the process of understanding a piece of text. In many real world applications of text mining, the context information can serve as an important guidance for understanding, analyzing, and mining the text data. For example, analyzing the search logs for contextual patterns can help a search engine developer to better serve its customers by re-organizing the search results according to the contexts of a new query. Analyzing the evolution of topics or decaying of topics in scientific literature would also help researchers to better organize and summarize the literature and to discover and predict new research trends. Also, analyzing the sentiments in customer reviews related to products and social events would help in summarizing the public opinion about them. Studying author-topic patterns can also make easy the finding of experts and their perceptive of the research communities.

Unfortunately, the importance of context in text mining has not been explored much. In most existing text information management systems, the importance of context is neglected. For example, search engines are considered as the most helpful tools to help users to find and access text information. However, none of the major search engines returns a webpage about 'Theory of Computer Science' on the first page when the query 'TCS' is sent by a researcher from the ACM conference.

There are different taxonomies of context from different disciplines. For example, the linguistic context which refers to the local surrounding text of a linguistic unit that is useful for inferring the meaning of the linguistic unit. Social context refers to the social variables that influence the use of language of a social identity (e.g., an author). A more general notion of context is known as the situational context, or context of situation, which is first proposed by the Polish anthropologist Bronis law Malinowski and then formalized with linguistic theory by J.R. Firth. It is concerned with the evaluable conditions or situations in which the text content is produced, including the situations that are either environmental or personal. Among the particular types of context, linguistic context is utilized in natural language processing to extend the feature space for supervised learning tasks such as tagging and parsing, entity extraction, and semantic role labeling. It is also used directly to solve problems such as word sense disambiguation and word clustering, in a way that the meaning of language units is compared through the comparison of their local linguistic context. The exploration of the more general context, context of situation, is quite limited in existing text mining work.

A basic assumption about text data is that the contents of the text usually cover a set of multiple topics where each topic is an implicit context. Understanding how to define and characterize context is a subject of research. Context learning research has started to evolve recently. By context we mean, any information that can be used to exemplify the situation inherent in the text.

◇ 5.2 CORPUS REPRESENTATION

If we look at the information on the web, around 80 per cent of the data is the textual data. There is a need to represent this corpus of large collection of text data into a form effective for retrieval of data. Let us assume that the corpus is a collection of text documents. Vector Space Model (VSM) is the most popularly used text representation model. It is an algebraic model for representing text documents. Using VSM, the text documents can be represented as vectors in feature space where features are the terms occurring in the documents.

For a corpus $D = \{d_1, d_2, ..., d_m\}$, the documents d_1 to d_m are represented as vectors in the feature space, as follows:

$d_j = \{w_{1,j}, w_{2,j}, ..., w_{t,j}\}$ where $w_{1,j}, w_{2,j}, ..., w_{t,j}$ represents word features of the document j and $w_{1,j}$ defines weight of $term_1$ of document d_j.

The N-dimensional feature space for M-documents is represented in matrix form as follows:

	f_1	f_2	...	f_n
d_1	$w_{1,1}$	$w_{2,1}$	...	$w_{n,1}$
d_2	$w_{1,2}$	$w_{2,2}$	...	$w_{n,2}$
...	...	...	...	...
d_m	$w_{1,m}$	$w_{2,m}$	...	$w_{n,m}$

Each dimension of the document vector corresponds to a separate term. If the term occurs in the document, it's value is considered non-zero in the vector. The dimensionality of

the vector is the number of words in the vocabulary (the number of distinct words occurring in the corpus). The values for these terms, known as weights, can be computed using several different techniques like Term Frequency (TF), Term Frequency Inverse Document Frequency (TFIDF), Chi-square (χ^2), Information Gain (IG), etc. Usually, text data representation is done by performing two basic steps: feature extraction and feature selection using some weighting model. Feature extraction refers to identifying significant features which represent the text document, and feature selection using some weighting model refers to assigning some appropriate weight values to the identified significant features of the text document.

◇ 5.3 CONTEXT-BASED LEARNING

Context allows to convert the raw data into rich decision pointers. What is true in particular context may not be true in some other context. Hence, it is a challenge in front of us to derive context from the given scenario, especially dealing with huge information.

◇ 5.3.1 Exploiting Hyperlink Context

Exploiting hyperlink context means exploiting the information surrounding a link in an HTML document. It exploits relevant hints that are directly provided in the structure of the HTML documents which people build on the web. A high degree of accuracy is achieved by combining large number of such hints.

Haveliwala proposed a modified page ranking algorithm that is context-sensitive. Instead of calculating a single Page Rank score of a document from a single, generic PageRank vector that is independent to the query, a dynamic technique is proposed by which a set of PageRank vectors, each of which is representative of a topic or category, are used to provide context specific ranking of results. At retrieval time, the topic sensitive PageRank is calculated by using the set of PageRank vectors for the topic the query belongs to. Thus, the context of the query is used to bias the documents rank score. Improved retrieval accuracy is reported with minimal online processing overheads.

Another advantage of context-based indexing categorization is that it can be applied to multimedia material since it does not depend on the contents of the documents to be categorized. It also restructures the catalogue. A context-based technique for ad hoc retrieval of web documents is proposed in called Context Matching (CM) which captures query context and matches against term context to determine term significance and relevance. CM has introduced a complete new way of interpreting and using context for retrieval and proved significant with positive impact on retrieval accuracy. It outperformed some best results by over 10 per cent and baseline runs by over 41 per cent.

◇ 5.3.2 Exploiting Linguistic Context

The linguistic context is commonly used in linguistics which refer to the local surrounding text of a linguistic unit that is useful for inferring the meaning of the linguistic unit. Linguistic context is utilized in natural language processing to extend the feature space for supervised

learning tasks such as tagging and parsing, entity extraction, and semantic role labeling. It is also used directly to solve problems such as word sense disambiguation and word clustering, in a way that the meaning of language units is compared through the comparison of their local linguistic context.

Relation Extraction

While exploiting context, we try to find out relationship between the entities talking about the text. So, information extraction is a technique which allows you to extract data from unstructured text about a particular domain which has multiple applications like summarization, question-answering, information retrieval, etc. Consider a literature document containing information about books, their respective authors, publication date and other related data. We might be interested in the relation between books and authors. Given a particular book, we would like to be able to identify the author who has written it; conversely, given an author's name, we would like to discover which books he/she has written. This involves Named Entity Recognition (NER) which includes the task of extracting entities from text and dividing them into different categories like:

- Person names (people names)
- Organization names (affiliation, administrative organizations, councils)
- Places (metropolis, nations)
- Date and time (different formats of date and time)
- Others (occasion, period, number, per cent, job title, etc.)

NER plays a crucial role in many linguistics processing projects like reference resolution, meaning representation, question answering, summarization, news searching, etc. Other NER-specific applications include question answering, summarization, news searching, etc.

Relation Extraction is one of the subtasks of Information Extraction. It defines a semantic relationship between two or more entities in a given text. For instance, Dr. Abdul Kalam Azad was the President of India. In this sentence, there are two entities, Person (Dr. Abdul Kalam Azad) and Location (India) and the relation (President) exists between these two entities. This concept of Relation Extraction can be used in various day-to-day applications like Library management, Resume Selection Process, applications related to medical domain, etc. There are various methods available for extracting these relations from open domain text or text related to particular domains like medical, sports, bollywood, Wikipedia pages, etc. Generally, a relation is considered as a triplet (Entity1, Relation, Entity2). For instance, Mark Zuckerberg is the CEO of Facebook. In this sentence, Mark Zuckerberg and Facebook are the two entities and CEO is the relation between them. So, the triplet can be written as **Mark Zuckerberg, CEO, Facebook**. Now, the question is, a text document might have numerous relations in it and nobody would be interested in all the relations from the document. However, according to the application of interest, extracted relations also vary. If the user is building an application for library, she/he might be interested in relations like Book-Author, Book-Date, etc., while in medical domain, relations like protein-protein interaction, drug-disease has more significance. Thus, depending upon what type of application the user is interested in, corresponding relations are extracted. Architecture of Relation Extraction is shown in Figure 5.2.

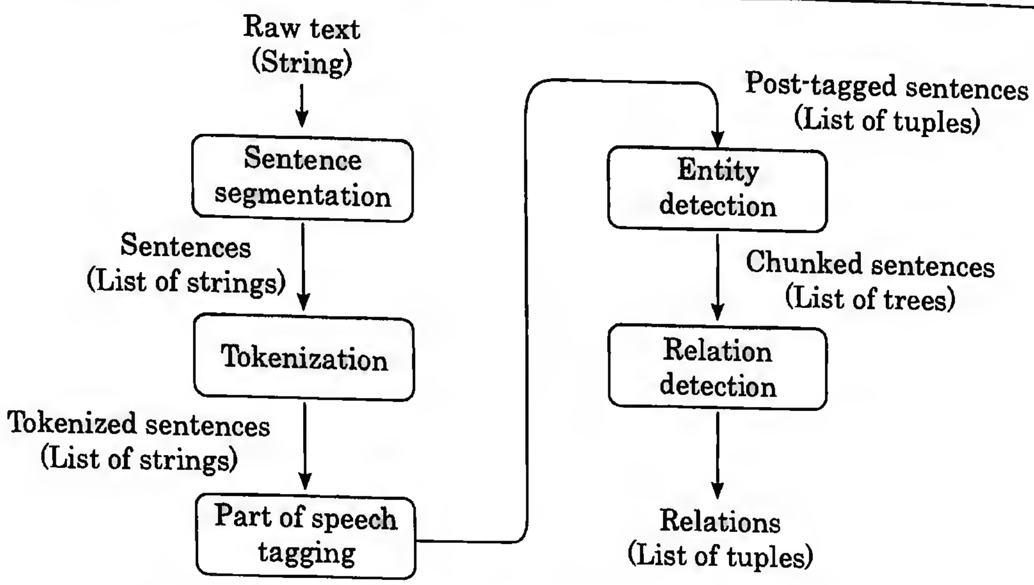


Figure 5.2 Relation extraction architecture.

GATE tool

GATE (General Architecture for Text Engineering) is an open source tool used for Information extraction and is also used by many industries. Many industries have million business dependencies on GATE. It extracts entities but relation extraction is not supported to a usable extend in GATE. Moreover, Named Entities that GATE extracts are very few in number. More the number of entities extracted, more the relations can be extracted further as relation extraction takes entity extraction output as input to that module. Further, Relation extraction plays a very crucial role in information extraction system for getting theme of the document. So, if Relation extraction is made stronger in GATE then it might add up to GATE to get more precise results in Relation extraction applications.

Adding more entities in GATE will increase the precision of the GATE tool. So, the main motivation behind this part of the chapter is to contribute towards the most used open source tool, like GATE in Information Extraction field by adding a module of relation extraction in it.

Please refer Annexure II for GATE Introduction, Installing and Running GATE, Features of GATE, Important Terms and Definition, Running GATE IDE, GATE Embedded.

ANNIE

An IE system, called ANNIE, is included in GATE as a plugin, a nearly new Information Extraction system (developed by Hamish Cunningham, Valentin Tablan, Diana Maynard, Kalina Bontcheva, Marin Dimitrov and others). Finite state algorithms and the JAPE language are used by ANNIE for various language processing tasks. It comprises a set of modules, discussed in the forthcoming sections.

Figure 5.3 shows working of ANNIE.

1. **Document reset:** This option allows to reset the document to its original state by removing all the tagged annotations.
2. **Tokenizer:** Tokenizer breaks a sentence into small parts like spaces, words, punctuations, symbols, sentences etc. Each token has different attributes like Category (NNP, PRP...), Kind, Orth (uppercase, lowercase...), etc.

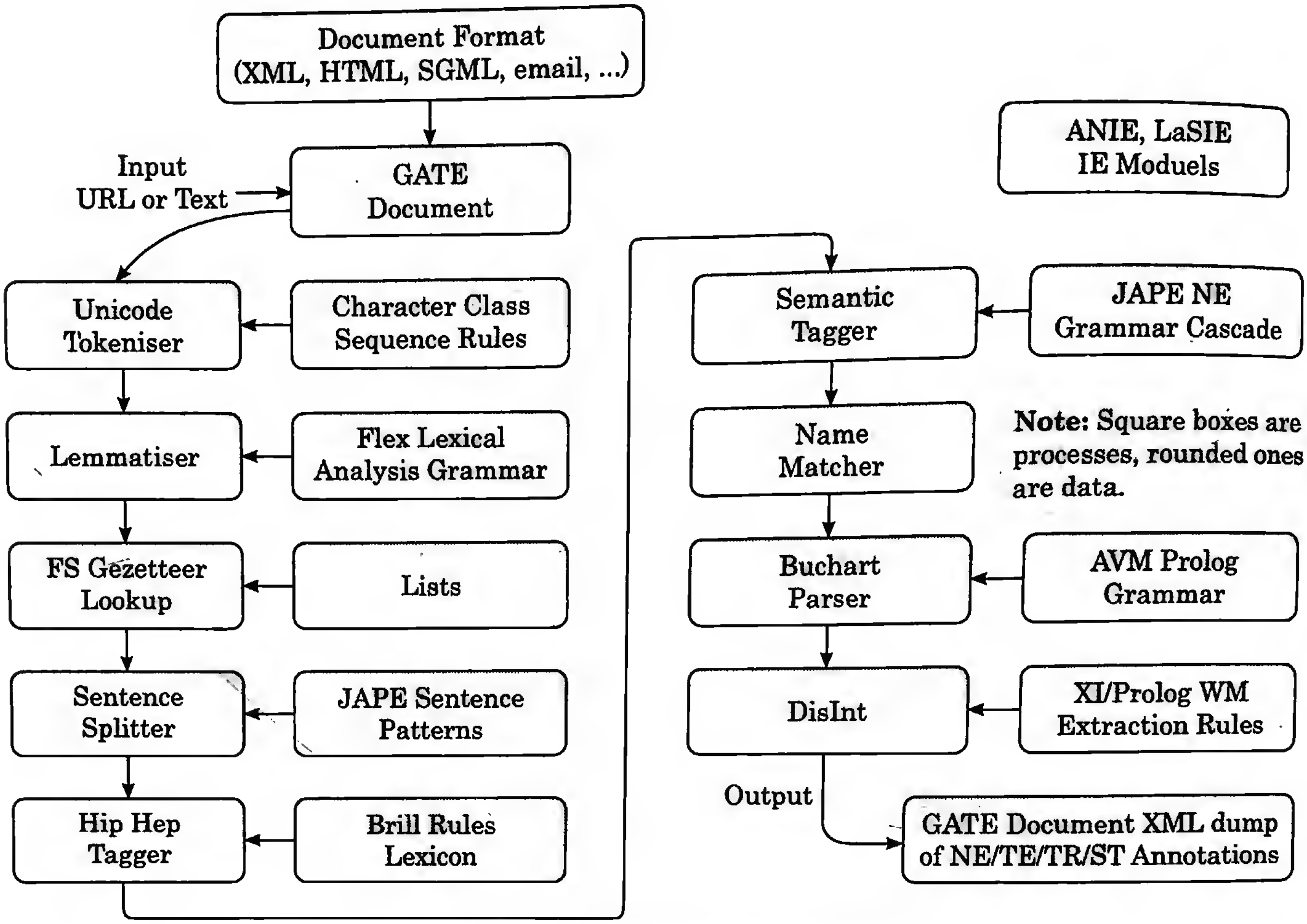


Figure 5.3 Working of ANNIE.

Tokenizer rules: Tokenizer has left and right hand side to it, in which LHS consists of the matched regular expression from the input, and RHS tells what annotation to include in output for the LHS and → is used to separate LHS and RHS.

The operators that can be used on the LHS are shown in Table 5.1.

Table 5.1 Operators and their meaning

Symbol	*Meaning*
|	Or
*	0 or more occurrences
?	0 or 1 occurrences
+	1 or more occurrences

Example: LHS → Annotation type; Attribute 1 = Value 1; Attribute *n* = Value *n*

Token types: Types of Token are as follows:

1. Words
2. Number
3. Symbol

4. Punctuation
5. Space token

3. **Gazetteer:** The gazetteer lists consist of files with .lst extension. Each file consists of a database for a particular entity like person, organization, etc. It is mainly used for extracting entities which are proper nouns.

Below is the example of units of currency (currency.lst):

Table 5.2 currency.lst

Cent
Penny
Luma
Paisa
Euro
Dollar

Lipa: A lists.def file is a directory of all .lst files used to access all .lst files. Each line of the .def file consists of following attributes:

- .lst file name,
- Major type,
- Minor type (optional),
- Language
- Annotation type (the name to be displayed in the list in GATE IDE).

The format of each line in the .def file is as follows:

(.lst file name) : (major type) : (minor type) : (language) : (annotation type)

Init Time parameters

(a) **listsURL:** lists.def file that contains mapping of all the .lst files is pointed by this URL.

(b) **encoding:** It defines the character encoding used while reading the pattern lists.

(c) **caseSensitive:** It defines whether the gazetteer should be case sensitive or not during matching.

Run-time parameters

(a) **Document:** The input files to be processed.

(b) **Annotation set name:** This is nothing but the name of the annotation set in which resulting Lookup annotations get created.

4. **Sentence Splitter:** The default splitter finds sentences based on Tokens. It creates Sentence annotations and Split annotations on the sentence delimiters. It uses a

gazetteer of abbreviations, etc. and a set of JAPE grammars which find sentence delimiters and then annotate sentences and splits.

5. **POS Tagger:** It requires Tokenizer and Sentence Splitter to be run first. It adds category feature to Token annotations. Each word or symbol gets annotated as a POS tag. Appendix B gives the list of all POS tags.
6. **Co-reference Tagger:** Different expressions may refer to the same entity. Orthographic co-reference module (orthomatcher) matches proper names and their variants in a document. For instance, [Mr. Smith] and [John Smith] will be matched as the same person.

◇ 5.4 GATE JAPE RULES

JAPE stands for 'Java Annotation Patterns Engine'. It uses finite state automata based on regular expressions for annotating entities and relations. It is written to find patterns in the given text document so that desired relations for which rules are written are extracted. It is also used for Entity Extraction. It works on regular language (not graphs) having strings consisting of letter, words, punctuations and alpha-numeric characters.

Entity Extraction using JAPE rule

Following is a stepwise discussion about writing a JAPE file for extracting book entity.

Step 1

- Go to GATE/Plugins/ANNIE/resources/gazetteer and write a book.lst file here.
- book.lst will consist of book names (one name per line).

Step 2

- Go to GATE/Plugins/ANNIE/resources/gazetteer and make changes to lists.def file.
- For each list, it denotes major type, a minor type (optional), a language and an annotation type. All of these parameters are separated by colons.

(name of .lst file) : (MajorType) : (MinorType) : (Language) : (Annotation Type)

- Lookup annotations are created by processing resource of the ANNIE gazetteer by default.
- Features like major, minor type and language get added to Lookup annotations and are used for pattern matching while writing the JAPE file. (We will see it in detail in Step 3).
- Open the lists.def file and add a new entry (line) here.
  book.lst : book : book : English : Book
  (For more details refer http://gate.ac.uk/sale/tao/splitch13.html)

Step 3

Go to GATE/Plugins/ANNIE/resources/NE and write a book.jape file at this location, of the form:

```
Phase: Book
Input: Lookup Token
Options: control = applet
Rule: Book
Priority: 25
(
Lookup.majorType = = book
)
:temp → :temp.Book = rule = "Book"
```

Phases combine to create a grammar, and each phase consists of multiple rules. Priorities can be assigned to rules within a same phase.

- Each JAPE file must contain a set of headers at the top, of the form:

```
Phase : University
Input: Token Lookup
Options: control = applet
```

- These headers are applied to all rules within that grammar phase. They contain Phase name, set of Input annotations and other Options.
- The Input Annotations list contains a list of all the annotation types you want to use for matching on the LHS of rules in that grammar phase.

  For example. Input: Token Lookup

  If no input is included, then all annotations are used.
- The matching style defines how we deal with annotations that overlap, or where multiple matches are possible for a particular sequence.

  Options: control = appelt.

  Different possible control styles are as follows:

1. appelt (longest match, plus explicit priorities)
2. first (shortest match fires)
3. once (shortest match fires, and all matching stops)
4. brill (fire every match that applies
5. all (all possible matches, starting from each offset in turn)

- Lookup.majorType == book

  This matches and accepts Lookup annotations whose majorType is book (For book. lst, we have defined majorType as book in the lists.def file).
- temp.Book, here temp is used to give temporary labels to the annotations obtained which satisfy the rule, and Book defines the rule name to be used for annotating the obtained Lookups.

  (For more details refer http://gate.ac.uk/sale/talks/gate-course-may10/track-1/module-3-jape/module-3-jape.pdf).

Step 4

This step is optional for java code. It is required while working with GUI.

- Go to GATE/Plugins/ANNIE/resources/schema and write a .xml file by referring to old ones.

Step 5

Most important

- Go to GATE/Plugins/ANNIE/resources/NE/main.jape
- Add the name of your JAPE file (Phase name) in the main.jape file.

Relation Extraction using JAPE rules

A case study illustrating extraction of organization-location relation is given below.

```
Phase:OL
Input:Token Organization Location
Options:control = appelt
Rule:OL
(
//Microsoft is located in Washington.
({Organization})
({Token})[0,7]
({Token.string = = "based"}|{Token.string = = "located"}|{Token.string = = "established"}|
    {Token.string = = "settled"}|{Token.string = = "headquartered"})
({Token})[0,3]
({Location})
)
:temp -> :temp.OL = {rule = "OL"}
```

In this code, organization and location annotations (which are already obtained) are taken as Input. Each word is referred to as token. We are using different patterns and keywords that can appear in the same sentence between these two entities (organization and location.) The operators that can be used on the LHS are as given in Table 5.1.

Entities which are proper nouns like person, organization, location, author, disease, etc., are extracted by using database in the form of .lst files. Other entities like date, currency, CGPA, per cent, etc., are extracted by writing regular expressions. For instance, the rule for CGPA entity is explained as follows:

```
Rule:CGPA
(
({Token.kind = = number_cgpa})
({Token.string = = "."})
({Token.kind = = number, Token.length = = "1"} | {Token.kind = = number, Token.length = = "2"})
)
:temp → :temp.CGPA = {kind = "number", rule = "CGPA"}
```

In this example, the grammar rule is written such that any number up to 2 decimal points is accepted. number_cgpa is a major type for the number_cgpa.lst file which contains numbers from 0 to 9 only. For such small set of numbers, you can also write the numbers in the rule itself instead of creating a .lst file, but since that is a standard way to write JAPE rules, the same method is followed for all the database related entities. Similarly, number in this example represents major type of the number.lst file which contains all numbers from 0 to mentioned level.

Consider another example for extracting Causal event relations.

```
Phase: Causal
Input: Token Sentence Date Time
Options: Control = appelt
Rule: Causal
(
(
({Token.string != "." })*
({Token.string = = "caused"} | {Token.string = = "causes"} | {Token.string = = "cause"} |
(({Token.string = = "result"} | {Token.string = = "results"} | {Token.string = = "consequence"} |
{Token.string = = "outcome"} | { Token.string = = "effect"} | { Token.string = = "upshot"} |
{ Token.string = = "outturn"} )
( { Token.string = = "into"} | { Token.string = = "of"})))
({Token.string != "."})*
) |
(({ Token.string ! = "."} *
({Token.string = = "because"})
({Token.string ! = "."} *
) |
(({ Token.string = = "If"} | {Token.string = = "if"} )
({ Token.string ! = "."}) *
({ Token.string = = ","} | {Token.string = = "then"} )
({ Token.string ! = "."}) *
) |
// This happened long after Zarah left the house
(({ Token.string ! = "."} *
({Token.category ! = "RB"}) | {Token.category != "RBR"} | {Token.category != "RBS"} |
    {Token.category != "RP"})
({Token.string = = "after"})
({Token.string != "."})*
) |
// He was getting over confident about his results as if he was the only one to participate
(({Token.string != "."})*
({Token.string = = "as"} | { Token.string = ="As"})
({Token.category != "IN"})
(({Token.string != "."})*
) |
// Since 1998, he was working in the field of cinematography.
(({Token.string != "."})*
({Token.string = = "since"} | { Token.string = ="Since"})
({!Date}{!Time})
({Token.string != "."})*
))
:temp  :temp.Causal = { rule = "Causal" }
```

There are numerous kinds of causal events those might occur in linguistic data, but this chapter considers a few cases to generate a basic causal event extraction model. The given example contains grammar rules for following cases:

1. If a sentence contains keywords like cause, result, outcome, effect, etc., then it is considered as a causal event.
2. Statements containing because keyword.
3. Statements having If – then clause.
4. Normally, a statement having 'adverb + after' almost falls in a temporal relation not a causation, so we have taken negation of it, so that causal events get extracted.
5. 'As + proposition' can hardly indicate a causation, so negation is taken into consideration.
6. Sentences having 'Since' keyword followed by a date or time event are not considered as causal relations. This rule is written for increasing the precision of the first rule.

Figure 5.4 shows the screenshot of entity and relation extraction model after extracting entities and Figure 5.5 shows the screenshot of entity and relation extraction model after extracting relations.

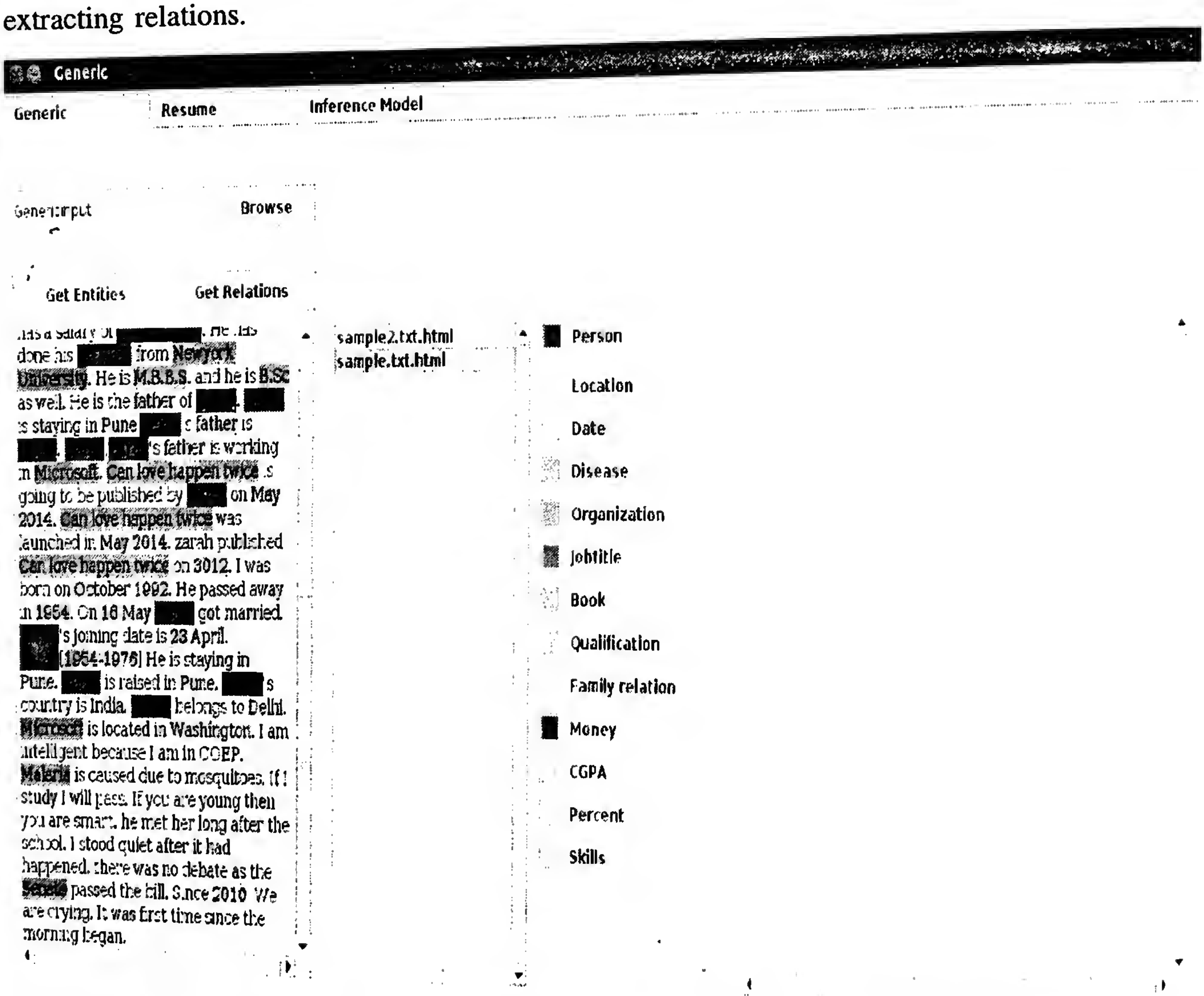


Figure 5.4 Entity and relation extraction model extracting entities.

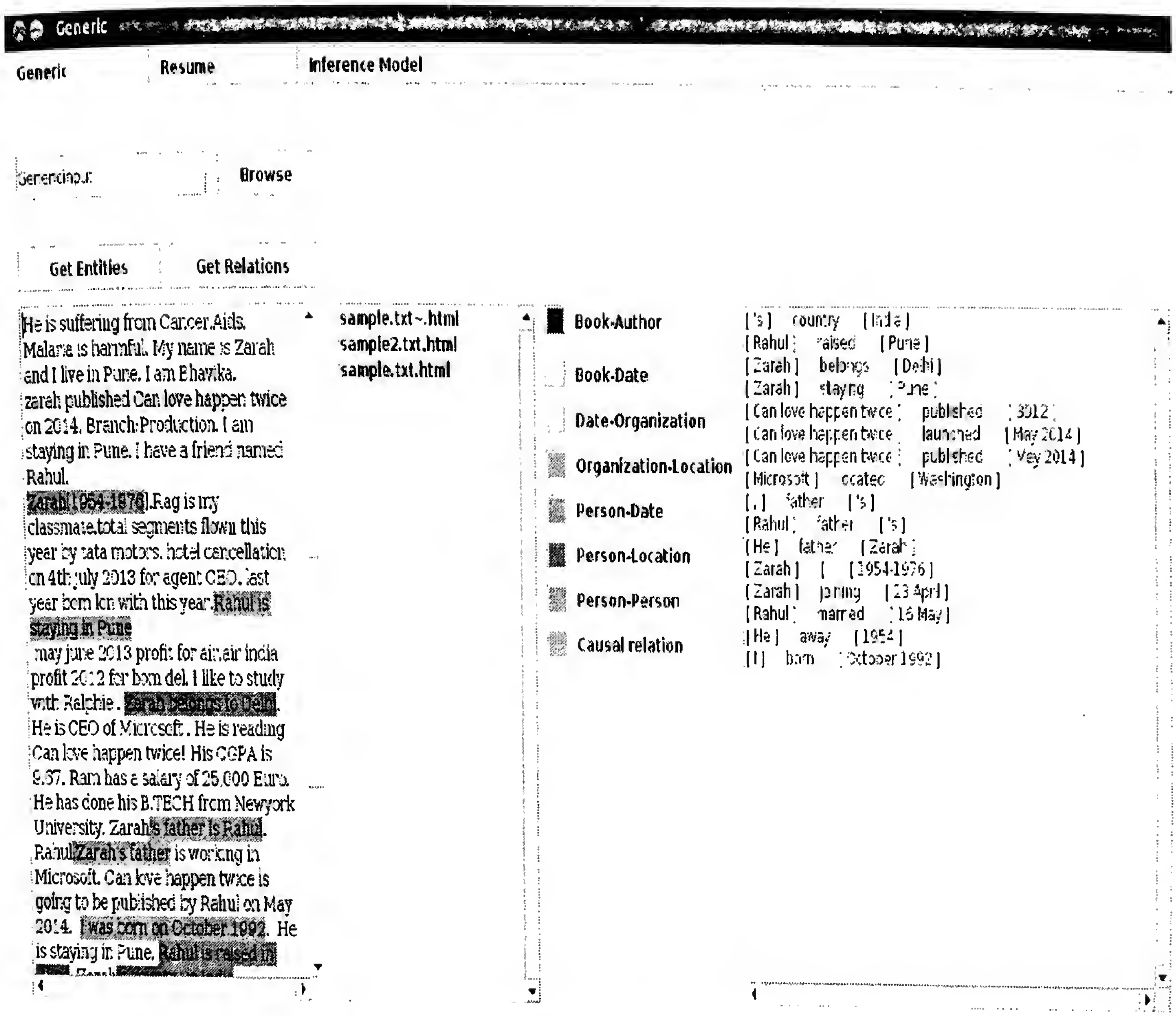


Figure 5.5 Entity and relation extraction model extracting relations.

◇ 5.5 TOPIC MODELLING

In topic modelling literature, topics are referred to as hidden patterns or short descriptions of documents in a text corpus. Technically, 'topics are semantically related clusters of words' which are used as a bridge between words and entities (e.g., documents or authors) to find hidden associations between them. A topic is informally defined as 'an underlying semantic theme; a document consisting of large number of words might be concisely modelled as deriving from smaller number of topics'.

Topic models are based on the idea that the documents can be represented as a mixture of topics. Generally speaking, the process for finding latent topics from text corpora by using topic models is called *topic modelling*. Technically speaking, it is the process of finding a topic z in a document d with defined probability distribution of words in a vocabulary V by using topic models.

5.5.1 Latent Semantic Analysis

We begin with Latent Semantic Analysis (LSA) for finding topics in textual data. It is also called Latent Semantic Indexing (LSI) when used in the context of information retrieval. The basic idea in LSA is mapping higher dimensional term-frequency vectors to low dimensional representation, called latent semantic space. This helps to provide more information than just occurrences of words in a document. The end goal is representation of semantic relations between words and/or documents in terms of their closeness in semantic space. Due to its generalization, LSA has proved to be a strong analysis tool. LSA uses Singular Value Decomposition (SVD), a technique based on matrix operations related to decomposition of eigenvector and performing factor analysis.

But LSI suffers from few drawbacks such as computational cost involved in getting the SVD, large memory resources required, and re-computation of the whole decomposition for inclusion of new documents.

Despite its success, there are number of shortcomings of LSA, such as computational cost involved in getting the SVD, large memory resources required, and re-computation of the whole decomposition for inclusion of new documents. On conceptual level, the representation obtained by LSA is unable to handle polysemy.

To handle issues of LSA, PLSA is proposed as the first probabilistic methodology with a latent layer and a strong statistical foundation, which is used for topic modelling.

5.5.2 Probabilistic Latent Semantic Analysis

The core of PLSA is a statistical model which is called the aspect model. Probabilistic Latent Semantic Analysis (PLSA) aims at identifying and distinguishing between different contexts of word usage without recourse to a dictionary or thesaurus. This has at least two important implications: Firstly, it allows us to disambiguate polysemy, i.e., words with multiple meanings, and essentially every word is polysemous. Secondly, it reveals topical similarities by grouping together words that are part of a common context.

The basic probabilistic topic model is known as probabilistic latent semantic analysis. It is also called Probabilistic Latent Semantic Indexing (PLSI) when used in the context of information retrieval. The topic model works on the basic assumption that there are k latent topics in the text collection, each topic is represented by a multinomial distribution over words. We use θ_j to denote the multinomial distribution for jth topic, over all $w \in V$. We introduce a new parameter $(\theta_j|d)$ to denote the distribution of selecting a particular topic from the mixture model by a document. $\{p(\theta_j|d)\}_{j=1\ldots k}$ thus makes a multinomial distribution of topics given a particular document. This distribution is sensitive to individual documents. The log likelihood function of D can then be rewritten as follows:

$$\log p(D|M) \propto \sum_{d\in D} \sum_{w\in d} \log \sum_{j=1}^{k} p(\theta_j|d)\, p(w|\theta_j)$$

where, w denotes a word token in the text collection, D denotes the document collection, M denotes the language model. One simply extension to PLSA is to mix the topics with a global background B. This will give us a modified PLSA model as follows:

$$\log p(D|M) \propto \sum_{d\in D}\sum_{w\in d}\log\left[(1-\lambda_B)\sum_{j=1}^{k} p(\theta_j|d)p(w|\theta_j) + \lambda_B\, p(w|B)\right]$$

The advantage of this model is that the common words in English, such as stop words and syntactic words, will be explained by the background context B. Therefore, the words with high $p(w|\theta_j)$ in each topic model will be meaningful content words through which we can interpret the semantics of the topic. λ_B is used for noise reducing benefits of model averaging. This modified PLSA model has been proven to perform well in text mining tasks.

As a special case, this includes synonyms, i.e., words with identical or almost identical meaning but it faces from two major drawbacks. First, the number of parameters in the model grows linearly with the size of the corpus, which leads to serious problems of over fitting, and second, it is not clear that how to assign probability to a document outside of the training set. It is generative at the words level but not at documents level. A model based on the unigram model was proposed, called Mixture of Unigrams. In unigram model, the words of every document are drawn independently from a single multinomial distribution. If the unigram model is augmented with a discrete random topic variable z, a mixture of unigrams model is obtained. Mixture of unigrams is based on the supposition that each document exhibits only one topic, which was too limited to effectively model text corpora. In order to overcome the limitations of PLSA, a generative probabilistic topic model, called Latent Dirichlet Allocation (LDA) was proposed.

◇ 5.5.3 Latent Dirichlet Allocation (LDA)

LDA provides generative models that explain how documents are created. It describes how each document obtains its words. Each hidden topic actually goes for building words for document. LDA assumes prior topic distribution in the document as well as distribution of words over topics. In LDA, a document can generate more than one topic, and it is possible to assign probability to documents outside the corpus by using variational inference algorithm and Gibbs sampling. It is generative at both words and documents level. LDA is computationally efficient than PLSA due to not having the problem of large parameters growth with the scale of input data.

PLSA is widely used in the context of text mining and information retrieval. One criticism of PLSA is that it has quite a lot of free parameters, so the model is likely to be over fit the data. An approach proposed to solve this is to introduce an additional regularization to the mixture coefficients, so that each multinomial vector is sampled from the same Dirichlet distribution. The new likelihood function of the collection can thus be written as follows:

$$\log p(D|M) \propto \sum_{d\in D}\int_{\vec{a}_d} p(\vec{a}_d|\alpha)\left[\sum_{w\in V}\log\sum_{j=1}^{k} a_{d_j}\cdot p(w\,|\theta_j)\right]d\vec{a}_d$$

Such a model is known as the latent dirichlet allocation in the machine learning literature. Please note that because of the integral, the parameter estimation for LDA cannot be handled by a standard EM algorithm. A more complicated estimation method is needed, such as variational

inference, Gibbs sampling, or expectation propagation. There are many other topic models, either extending PLSA or extending LDA.

◇ 5.6 SITUATION MODELLING

Situation models extract information that characterizes the situation inherent in the text. Researchers proposed that understanding any text involves constructing a mental representation of the text itself. A situation model is user-perspective comprehension of the text. Thus, situation models are mental representations of various entity associations described in a text rather than of the text itself. They are mental representations of the people's understanding about objects, locations, events, and actions described in a text. From this perspective, any information that can be used to exemplify the situation will build a context.

The model of situation building aims at building a situation from text by first determining similarity between two sentences. Then the similarity value is used for determining the coherence between the two sentences. This helps in forming chunks of text sentences where they are similar and coherent. The similarity is tested on four different levels: syntactic similarity, semantic similarity, similarity by co-occurrence and similarity by grammatical relations, and the coherence is checked by comparing with predefined threshold. Situations are extracted for every text chunk.

Coherence in linguistics is what makes a text meaningful semantically. Coherence is achieved through syntactical features such as features that directly trace the act of utterance, features that can be used as a regular grammatical substitute for some preceding word or group of words, as well as pre-suppositions and implication connected to general world knowledge. To find the coherence values between sentences, we require word similarity which helps in calculating the sentence similarity. We have used the knowledge base, WordNet, for the same. The sentence similarity is found by determining word similarities between the sentences. There are different types of similarities based on which two or more sentences can be distinguished like syntactic similarity, semantic similarity, similarity by co-occurrence, and similarity by grammatical relations. In syntactic similarity, syntactically same words are given high similarity value. If the sentences contain same words, then sentence similarity is influenced to a great extent. In semantic similarity, the words which may not be syntactically same but may be semantically similar are given high similarity value. In similarity by co-occurrence, words which co-occur repeatedly with similar words have increased similarity value which helps in increasing the sentence similarity. In similarity of grammatical relation, words which occur repeatedly with the similar grammatical relations are considered to be possibly similar, and therefore, the similarity value of such words is increased by some factor, thereby increasing the sentence similarity. Thus, we find similarity between two consecutive sentences which is compared with a predefined threshold. Whenever the similarity value goes below the threshold value, we can say that the text has a break in coherence. We form chunks of text using this strategy. Then for each chunk, we extract a situation.

Situation extractor aims at finding important parts from the chunks of text. It uses the score values of sentences, and outputs the comprehension of the text. The score for each sentence is calculated using the local and global score values of each word within the sentence. The local

score of a word is calculated by adding the score of the word with the score of the clause in which the word appears. The score given to a word is the frequency of the word and the score of the clause is calculated using the aggregated score of the words in all the trigrams containing the word.

◇ 5.6.1 Determining Coherence

Coherence in linguistics is understood to be semantically meaningful text. In this section, we find the coherence values between sentences.

To find the values, we require word similarity so that we could calculate the sentence similarity. We have used the knowledge base, **WordNet**, for the same.

WordNet[1]

WordNet is a lexical database developed by George Miller at the Cognitive Science Laboratory at Princeton University. It contains a very large collection of English language words structured as a semantic network, with nodes as terms linked with IS-A relation. In WordNet, words are grouped together into sets of synonyms, called Synsets, each expressing a distinct concept. A Synset contains a concise definition of each word, called gloss definition for all senses associated with the word. WordNet follows different grammatical rules for distinguishing between nouns, adjectives, verbs and adverbs. Prepositions, determiners, etc. are not supported in WordNet.

To find similarity between words, more specifically path similarity, WordNet is used. Path similarity measure is based on determining the shortest path between the word senses in a IS-A hierarchy of WordNet. It returns a score depending on how similar the two word senses are. The score ranges between 0 and 1, except all cases where a path cannot be determined or found, in which case, none is returned. A score of 1 indicates complete identity, for example, comparing a sense with itself will return 1.

Path-based measure

An intuitive method to measure the semantic relatedness of word senses using WordNet, given its tree-like structure, would be to count up the number of links between the two synsets. The shorter the length of the path between them, the more related they are considered. Such a measure had been experimented with by Rada et al. for measuring semantic relatedness of medical terms, using a medical taxonomy, called MeSH. Their measure performed rather well. A measure suggested by Leacock and Chodorow does almost this, using WordNet. The measure suggested by Leacock and Chodorow considers only the IS-A hierarchies of nouns in WordNet. Since, only noun hierarchies are considered, this measure is restricted to finding relatedness between noun concepts. The noun hierarchies are all combined into a single hierarchy by imagining a single root node that subsumes all the noun hierarchies. This ensures that there exists a path between every pair of noun synsets in this single tree. To determine the semantic relatedness of two synsets, the shortest path between the two in the taxonomy

1. http://wordnet.princeton.edu

is determined and is scaled by the depth of the taxonomy. The following formula is used to compute semantic relatedness:

$$\text{Related}_{lch} = -\log\left[\frac{\text{Shortestpath }(c_1, c_2)}{2 \times D}\right]$$

where c_1 and c_2 represent two concepts, *shortestpath* (c_1, c_2) specifies the shortest path between two concepts c_1 and c_2, and D is maximum depth of taxonomy. This method works on the assumption that the weight of every path or link in the taxonomy will be the same. This assumption does not hold. It is experimented that concepts that are away by single link downward in the hierarchy are said to be closer or more related than concepts higher upward in the hierarchy. This approach works relatively well, in spite of its lack of complexity.

◇ 5.6.2 Determining Sentence Similarity

Sentence similarity is found by determining word similarities between the sentences. There are different types of similarities like syntactic similarity, semantic similarity, similarity by co-occurrence, and similarity by grammatical relations. We will be discussing all these types in the following section:

Types of similarities

1. **Similarity at syntactic level:** A similarity value of 1 is set for the words which are the same syntactically. If the two sentences target for determining similarity has maximum identical words, then similarity between these two sentences is said to be very large.

For example:

- This fruit is a red apple.
- This fruit is an apple.

The above sentences show large similarity value as there are lots of common words.

2. **Similarity at semantic level:** A similarity value of 1 is set for the words which are the same semantically. If the two sentences target for determining similarity has words which are not similar syntactically but are similar semantically, then the similarity value between such words is also set to 1.

   The similarity of semantically related words is given weightage while determining sentence similarity.

For example:

- Cooked the fish.
- Grilled the bass.

This example comprises semantically related words, such as (cooked, grilled) or (fish, bass). Thus, the example has maximum semantic similarity.

3. **Words co-occurrence similarity:** All those words which co-occur along with same set of words in the text, repeatedly are assumed to be of similar meaning. If the two sentences target for determining similarity has words which co-occur with same set of words frequently, then the similarity value of such words is increased by some factor, thereby, increasing the sentence similarity.

For example:

- Car met with an accident.
- Scooter met with an accident.

In the above sentences, there is a possibility that Car and Scooter are similar.

4. **Words grammatical relation similarity:** All those set of words which occur repeatedly across the text with the similar grammatical associations are assumed to be similar. This theory increases the similarity value of such grammatically similar words by some factor, thereby, increasing the entire sentence similarity.

For example:

- Abhay drove the motorcycle.
- Abhang rode the bus.

In the above sentences, motorcycle and bus are possibly similar as they come with similar grammatical relation.

◇ 5.6.3 Situation Building

After determining text chunks which are coherent, find the score of each sentence belonging to the chunks. So, we have text chunks with higher score values sentences. But while building a situation, higher sentence score is not the only criteria for adding a sentence in a situation. Initially, the highest score sentence is added to a situation. Before adding the next higher score value sentence, check to find if this considered sentence is connected with the first highest score sentence by a connective like AND. If so, then the next higher score sentence is found unnecessary and not added to situation. This is because the words like 'AND' may speak about things already spoken about and may not add any new information to the situation.

If the sentence selected is a simple sentence with the highest score value, then there is a high possibility of it speaking about a topic not spoken about before (as it is an important member of the chunk) and therefore gets selected from the text chunk. Hence, it is included straightaway in the situation. If the sentence starts with an elaborating connective, its importance is set to 0 by setting its score value to 0. If it does not start with an elaborating connective but has a connection to the previous sentence, then also its importance is decreased to 0, i.e., setting its score value to 0. The fate of such sentence depends on the previous sentence. Even if the sentence has lower importance then also its score value is set to 0. The current sentence considered is removed from the text chunk considered for building situation. This process is repeated till we achieve the required number of sentences in the situation which are coherent and highly similar. Thus, by applying this process on every chunk, we form situations of incoherent chunks, thereby, forming the complete situation.

◇ 5.7 BIG DATA AND TEXT CLASSIFICATION

◇ 5.7.1 Introduction

Big Data is currently defined using five data characteristics: volume, variety velocity, value and complexity. There are additional characteristics related to data, such as the fast growth of volume, variety, value, management, and security. It means that at some point in time, the current techniques and technologies may not be able to handle storage and processing of the data when the volume, variety and velocity of the data are increased. Each of the characteristics represents a serious problem of technical research, and is discussed below.

Big Data volume

The amount of data getting generated every other minute is very large. The data that is difficult to be processed using traditional tools and is in the form of petabytes of data requires large storages and would be ever increasing. This increase in data can be managed by purchasing additional storage, however such expenditure would be unreasonable.

Fast growth of data

The data that is increasing at a faster rate is the unstructured data. This data comprises of information such as photos, emails, Twitter tweets, data of Facebook, conversation records from call centers, movies, financial transactions, website clicks, datasets of medical records, images, documents, weather forecasting records, sensor data, text and many more. According to statistics, unstructured data is capturing more than 80 per cent of data in any organization. It is said to constitute nearly 80 per cent of worldwide data and comprising of 90 per cent Big Data. Much of the unstructured data is random and therefore not modeled and becomes difficult to analyze. Appropriate strategies need to be developed for managing such a huge data.

◇ 5.7.2 Management of Big Data

Today, in many organizations most of the data are stagnant. Data received from various resources, such as sensor network data, private and public data, log files, etc., are highly disorganized. Earlier most of the companies were not able to capture and store these data, and also existing traditional tools were incapable to analyze them in finite amount of time. However, the new paradigm of Big Data technology has shown great performance in many dimensions along with providing excellent decision-making support. The fundamental objective behind Big Data technology is to minimize the hardware and computational cost and analyze the large pool of information available for effective decision making. Rightly managed Big Data are always available, consistent, secure and handy. Hence, many of the Big Data applications are useful in different scientific complex disciplines, including biology, biogeochemistry, medicine, genomics, astronomy, atmospheric science and many more.

Evolution of Big Data technology leads to management of very high volumes of data without requiring high cost supercomputers. There are tools and techniques available for effective data management, including Simple DB, Google BigTable, Not Only SQL (NoSQL), Voldemort, MemcacheDB and Data Stream Management System (DSMS). However, still special tools and techniques are to be developed for storing, accessing and analyzing large volumes of data in near future. Some of the popular tools and techniques for Big Data are Hadoop, MapReduce, and Big Table. These techniques have efficiently done data management by effectively processing huge volumes of data and that too in timely manner. It is also cost effective.

Hadoop

Hadoop is a framework for processing data in parallel using MapReduce pattern, where the entire work is divided into different tasks or blocks and gets distributed across group of machines (clusters). Currently, Hadoop is used on large volumes of data. With Hadoop framework, many of the enterprises are able to efficiently tackle data that were unmanageable and difficult to analyze previously.

Hadoop is composed of different components such as HBase, Kafka, HCatalog, Pig, Oozie, ZooKeeper, and Hive. However, the most widespread components are Hadoop Distributed File System (HDFS) and MapReduce.

Figure 5.6 illustrates the Ecosystem of Hadoop, with relationships among various components.

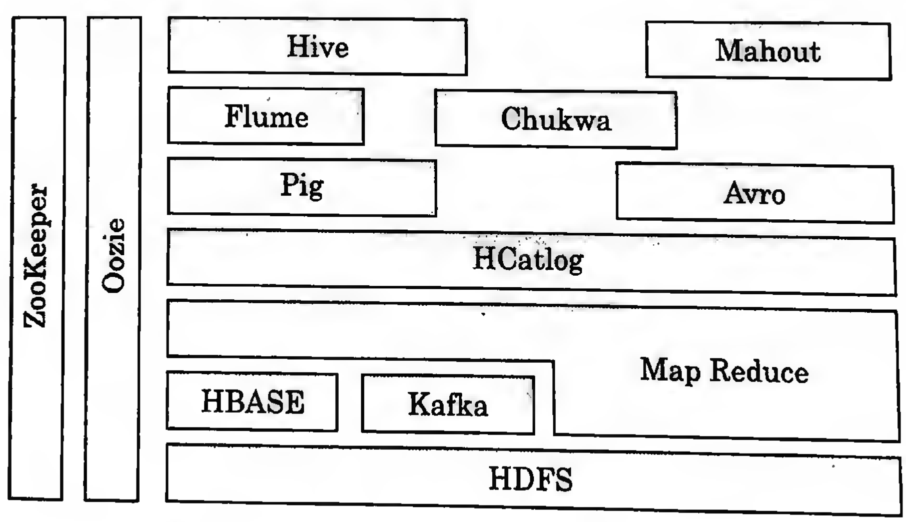


Figure 5.6 Hadoop ecosystem.

Hadoop Distributed File System (HDFS)

HDFS is planned to run on commodity hardware. It is highly fault tolerant and gives high throughput. HDFS supports a master/slave architecture. HDFS cluster comprises single Namenode, called master server and a number of Datanodes, called slave nodes. The Namenode handles the file system namespace and controls access to files by clients. A file is divided into one or more blocks (64 MB size) and HDFS stores these blocks in DataNodes. Replication of all HDFC files is done in multiples for facilitating parallel processing for the huge volumes of data.

HBase

HBase is a scalable data management store and modeled after Google's BigTable. This system is targeted to support column-based large tables, which speed up the performance. Access to HBase is all the way through Application Programming Interfaces (APIs) such as Java, Thrift, and REpresentational State Transfer (REST) which do not have their own scripting or query languages. Specifically, HBase depends fully on the instance of ZooKeeper.

ZooKeeper

ZooKeeper is a coordination service for configuration management, distributed synchronization, naming and group services. Historically, for each distributed application, many developers have to work hard to re-invent these services which was absolutely time consuming and more prone to errors, as correct implementation of these services was very difficult. Zookeeper made it easy and simple to implement these services and other primitives, and relieved the developers to focus more on semantics of application. It is the only distributed service which stores configuration information and has master as well as slave nodes.

HCatalog

HCatalog performs HDFS management. It is responsible for storing metadata information and generating tables for huge volumes of data. HCatalog relies on metastore of Hive which is integrated with added services using a data model. The added services include MapReduce and Pig. HCatalog can further be expanded to HBase using this data model. HCatalog is a data sharing source between tools and execution platforms. It simplifies user communication using HDFS data.

Hive

Hive is a SQL-like data warehouse infrastructure. It is built on top of Hadoop. HiveQL is its own query language compiled by MapReduce. Hive's design reflects its use for managing and querying structured data. Being focused on structured data, certain optimization and usability features can be added by Hive that MapReduce, being more general, does not have. Hive is based on three related data structures: partitions, tables, and buckets. HDFS directories resemble the tables which are distributed in various partitions and, ultimately, bucket.

Pig

Pig is an extension of Hadoop. It simplifies Hadoop programming by providing a high-level data processing language by maintaining scalability and reliability of Hadoop framework. Pig has its own compiler which compiles and runs the language script with respect to evaluation mechanism, which is Hadoop. Pig can operate on any data where the data can be relational, semi structured, unstructured or even nested. Pig supports this diverse data by providing complex data types, such as bags and tuples which help in forming refined data structures.

Mahout

Mahout provides machine-learning libraries and is highly scalable. To enable large-scale data processing, it has been tightly integrated with Hadoop MapReduce model. Mahout include implementations for number of machine learning algorithms, such as Naïve Bayes classification, clustering by *K*-means, recommendation engines, logistic regression model, random forest decision trees, and collaborative filtering.

Oozie

The management and execution of job flow are coordinated by Oozie in any Hadoop system. It is integrated with other Hadoop frameworks, such as Distcp Sqoop, Streaming MapReduce, Java MapReduce, Pig and Hive. Oozie uses Directed Acyclic graph (DAG) for arranging Hadoop tasks.

Avro

Avro framework supports data serialization and provides data exchange services required by Hadoop. There can be exchange of Big Data among different programs written in any programming language, using Avro. Data can be efficiently serialized into files or messages using the data serialization service. The data along with its definition is stored together in one message or file by Avro, making it simple for programs to understand the information getting stored in an Avro file or message, dynamically. Avro stores data in binary format, making it dense and efficient, whereas it stores the data definition in JSON format, thereby, making it convenient for reading and interpreting. Markers are included in Avro files for dividing large datasets into smaller subsets capable for MapReduce processing.

Chukwa

Chukwa is an open source framework for data collection and analysis. It inherits Hadoop's scalability and robustness as it is built on the top of HDFS and MapReduce framework. The data from distributed systems is collected and processed by Chukwa and then stored in Hadoop. Chukwa has been included as an independent module in the distribution of Apache Hadoop. It also includes a powerful toolkit for monitoring, displaying and analyzing results for better usage of the collected data.

Flume

Flume is typically used to collect, aggregate and move large amount of log data in and out of Hadoop. The flume architecture is simple, depending on data flow streaming. It has tunable reliability and recovery mechanism with robustness and fault tolerance. It has two channels, viz. sources and sinks. The system logs and Avro files are included in sources, whereas HDFS and Hbase are referred by sinks.

Table 5.2 summarizes the functionality of the various Hadoop components discussed above.

Table 5.2 Hadoop Components and their Functionalities

Sr. No.	*Hadoop component*	*Functions*
1.	HDFS	Storage and replication
2.	MapReduce	Distributed processing and fault tolerance
3.	HBASE	Fast read/Write access
4.	HCatalog	Metadata
5.	Pig	Scripting
6.	Hive	SQL
7.	Oozie	Workflow and scheduling
8.	ZooKeeper	Coordination
9.	Kafka	Messaging and data integration
10.	Mahout	Machine learning

In the Big Data research, the term Big Data Analytics is defined as the process of analyzing and understanding the characteristics of massive size datasets by extracting useful geometric and statistical patterns. *Ideally,* when the volume, variety and velocity of the data are increased, the current techniques and technologies stop functioning as expected within a given processing of time. Many applications suffer from the Big Data problem, including network traffic risk analysis, geospatial classification and business forecasting.

The new technologies can help to conduct Big Data analytics on various applications. The techniques, Hadoop Distributed File Systems (HDFS), Cloud technology and Hive database can be combined to address the problems like Big Data classification.

Nonetheless, many traditional techniques for text classification may still be used to process Big Data. Some representative methods of traditional text classification include SVM, Naive Bayes, Decision Trees, etc.

◇ SUMMARY

This chapter discussed the techniques for Big Data text categorization, topic Modelling and context-based learning. It gives Introduction to relation extraction and the GATE tool. The mathematical models LSA, PLSA, and LDA for topic Modelling are discussed with their drawbacks and limitations. Methodologies for building situations are discussed in Situation Modelling. In this, WordNet path-based measure is used for determining the semantic relatedness of two synsets for similar type words followed by determining sentence similarity of four types—syntactic similarity, semantic similarity, similarity by co-occurrence and similarity of grammatical relation. Finally, situations are extracted by determining coherent and in-coherent sentences from chunks of text which will result in complete situation.

Multiple Choice Questions (Select all if applicable)

1. In the multi-label text classification, a text document
   (a) belongs to just one class of a set of many classes
   (b) belongs to one class of a set of 2-classes
   (c) may belong to several classes of a set of many classes at the same time
   (d) None of the above
2. Exploiting Hyperlinks Context, means
   (a) exploiting local surrounding text information of a linguistic unit
   (b) exploiting relevant hints that are directly provided in the structure of the HTML documents
   (c) exploiting the information surrounding a link in an HTML document
   (d) exploiting text information spread across the hyperlink
3. Named Entity Recognition (NER) includes the task of
   (a) extracting person names (people names)
   (b) extracting Organization names (Affiliation, Administrative organizations, councils)
   (c) extracting places (Metropolis, Nations)
   (d) All of the above
4. LSA is
   (a) maps high-dimensional count vectors, such as term-frequency (tf) vectors arising in the vector space representation of text documents to a lower dimensional representation, called latent semantic space.
   (b) represents semantic relations between words and/or documents in terms of their proximity in the semantic space.
   (c) unable to handle polysemy.
   (d) All of the above
5. WordNet uses .... measure to find the semantic relatedness of word senses.
   (a) path-based (b) information content-based
   (c) gloss-based (d) Jiang Conrath

Concept Review Questions

1. What is text mining? Discuss a few applications of the same?
2. Explain text categorization and their paradigms.
3. What do you understand by context? Explain Context Learning. Also, discuss the different approaches for context-based learning.
4. Explain NAMED-ENTITY associations? Using GATE tool, write a JAPE rule for extracting causal event relations.
5. Explain topic Modelling. Also, explain situation Modelling.

6. Explain the knowledge base, WordNet, with its applications.
7. Discuss the five data characteristics of Big Data.
8. Explain the tools and techniques used for handling Big Data.

Critical Thinking Questions

1. How to build situation vectors for social media text data?
2. How to build context from these situation vectors?

Laboratory Assignments

1. **Problem Statement:** Smart Content Manager Application for Recommendation (either hotels, movies or shopping malls) based on Context.

Aim:

(a) Building user profiles for understanding type of user with his/her interest.
(b) Building ontology for hotels, movies and shopping malls as they are the recommendation types.
(c) Named-entity recognition of text messages.
(d) Building Relation Extraction model for extracting associations from text messages. Extraction of Context like location, date, time and user-type.
(e) Building inference model for recommendation.

Data objects: User profiles, data of location, date and time and Instant text messages.

Output: Recommendation to user for nearby hotels, movies or shopping malls based on context.

Challenge: Named-entity recognition of text messages. Extracting associations from short text messages, in continuous time-mode, at a particular location and date.

Methodologies suggested:

(a) Use of GATE tool for Named-entity recognition.
(b) Rule-mining algorithms for extracting associations from short text messages.
(c) Probabilistic model for inference theory.

2. **Problem Statement:** Topic based categorization (Categorization of tweets into three categories: positive tweets, negative tweets and neutral tweets.)

Aim: Tweet collection. pre-processing collected tweet data. Vector space data representation of tweet data. Feature extraction and selection for building positive and negative vocabulary. Probabilistic model for categorizing tweets.

Data objects: Tweets in form of text.

Output: Associating sentiment/category to each input tweet generated.

Challenge: Understanding positive, negative and neutral vocabulary for the tweets.

Methodologies Suggested:

1. Use of stemming, stop-word removal, tokenization for pre-processing.
2. Use of standard feature extraction and selection techniques like TF, TFIDF, etc.
3. Naïve Bayesian probabilistic methodology for inferring positive, negative or neural tweet.

6

Multi-label Big Data Mining

—Dr. Sonal Dharmadhikari
—Prof. Sheetal Sonawane

◊ 6.1 INTRODUCTION

Widespread use of internet led to large availability of textual information in the form of blogs, emails, downloaded papers, opinions of people through social media, online news articles, medical reports, annual reports of organization, etc. As an effect, text data has proven to be a major information source in small scale to large scale organizations. Text document is a multi-faceted object. Moreover, the unstructured nature of text often generates its ambiguous representation. The challenge of mining useful information from unstructured textual data is a top priority for organizations that are looking for efficient ways to search, sort, analyze and extract relevant information from large text collection they store and create daily. It is very difficult to process such large text collection gathered from various sources using traditional database and software methods. Big Data Analytics faces even more challenges while analyzing such unstructured, heterogeneous and large volume of textual data. It demands for its systematic organization and classification with the purview of efficient storage and retrieval in many applications like sentiment analysis, classification of emails, classification of news articles, Authorship generation, healthcare domain, opinion mining, web page classification, verdict predictions of election process, digital forensics, banking, security, etc.

It has been observed that these ambiguous text objects are often representing the property of multilabelility. The multi-label text document may simultaneously belong to more than one concept classes in the process of automated text classification. For example, in the process of automated classification of online news articles, the news about the scams in the commonwealth games in India may be classified into classes like sports, politics, country—India. Similarly, a research paper based on protein synthesis using data mining approach may represent the category of Bioinformatics, Computer, Biotechnology and Data Mining. Multi-label unstructured mining refers to the analysis of huge unstructured text document repository in order to extract meaningful information and most relevant class labels associated with each textual object. Various practices based on Statistics, Text Mining, Machine Learning, Information Retrieval and Natural Language Processing are utilized to achieve the aforesaid objective. The property

of association with more than one category makes the task of automated text classification more challenging. Hence, multi-label classification problems have been focused considerably and may play crucial role even in mining huge unstructured data in Big Data scenario.

Thus, incorporating multi-label learning aspect into Big Data analytics is the demand of the present era. The importance of concept and context in text analytics is described in previous chapter. In view of Big Data, handling of multi-label data is important for various applications. This chapter is intended to introduce the important phases in the life-cycle of multi-label unstructured mining. The crucial aspect of graph-based data representation and Modelling in the multi-label context is also highlighted through this.

◇ 6.2 PHASES IN MULTI-LABEL UNSTRUCTURED TEXT MINING

Multi-label problem has received significant attention in the machine learning, information retrieval and NLP-based research so far. The traditional methods to address this problem may not be applicable to Big Data analytics. Literature reveals that high dimensionality existing in feature and label space of text in Big Data analytics raises significant challenge to make multi-label methods suitable for Big Data analytics. Apart from the high dimensional feature space, the increasing number of labels, association between label set collections and heterogeneity between textual collections for being collected from varied sources makes the task of Big Data analytics even more complicated. Hence, the process of multi-label unstructured mining is divided into six phases as depicted in Figure 6.1. The phases are namely data collection, data processing, data cleaning and transformation, data representation and modelling, exploratory data analysis, validation and reporting decisions/predictions.

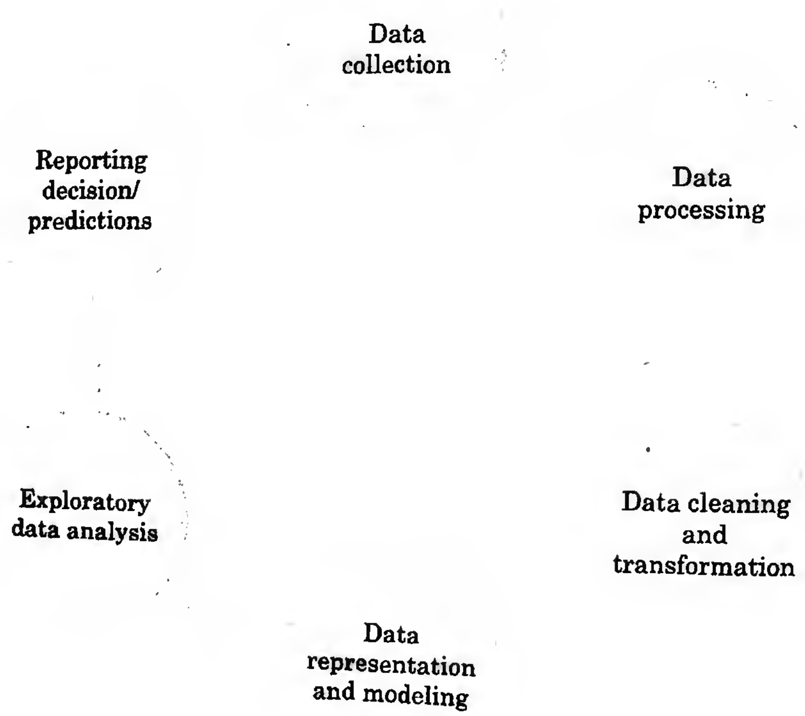


Figure 6.1 Phases of multi-label unstructured mining.

In the Big Data context data may be collected at different sources. Data collection phase is responsible to collect the data from all the different sources. The collected data is processed in order to represent in the form required by application which processes it. Data processing phase is responsible for this task. Being collected from variant sources, the textual data may get contaminated with noise, which is removed and then transformed into reduced form from the efficient storage and processing point of view by means of data cleaning and transformation phase. Subsequently, the textual data and labels are represented in a way so as to predict correct business decisions in the data representation and Modelling phase. Various machine learning algorithms are then applied on it for exploratory analysis. The decisions are evaluated for their correctness and conveyed by means of various comprehensive graphs and charts and further utilized for Business Intelligence process. Subsequent sections explore the aforementioned phases in detail.

Data collection

Performance of organizations involving Big Data is dependent on efficient ways of data collection, as business decisions are dependent on the gathered data to the large extent. For example, in Banking sector, the data used for the analysis is generally transactional, for example, customer's history for purchasing activity using debit card and credit card of the bank, loan payment history of customer, daily transactions performed by customer. Managers can ask questions such as to which customers the newly launched scheme of credit card should be intimated based on past record and get answers in real-time that can be used to help make short-term business decisions and long-term plans.

Big Data collection is a major activity among small to large business enterprises. Business intelligence is enhanced by means of optimized data collection process. There are varieties of ways by which organization gathers needed data. The data collection strategies of organization depend upon types of technologies being used by it.

Generally, many organizations collect the data available from their customers using the internet. The Big Data collection process may collect data using the internet technology, GPS system, mobile technology, call centre logs, social networking site access patterns, customer reviews and feedback, client requirements, etc. It is apparent for data scientists to integrate this data gathered from different sources in order to conduct Big Data analytics as data coming from multiple places within the organization need to share a common format for efficient processing. Just imagine that how various data sources can introduce serious inconsistencies such as variations in the characters allocated or data type used for customer names, format used to represent birth date of customers, using different currency units (e.g., dollars versus rupees), redundant customer information, etc.

Data processing

Data preprocessing step is generally employed on data prior to application of any machine learning algorithms. Text processing component in multi-label text mining is responsible to map the text document into the form which can be processed by subsequent phase. The document is generally represented in the form of feature vector. Obviously, huge number of features are generated as a outcome of data processing phase. Further, every feature has been assigned

with weight to describe the category labels. Indexing component assigns weight value to each feature. The unprocessed or poorly processed text collection may yield incorrect decisions. It has been observed that success of decision-making process is dependent largely on this phase. There exists Hadoop-based frameworks like hive, pig and mahout for data processing and are implemented in the map-reduce paradigm.

Data cleaning and transformation

As discussed previously, because of the large number of features generated in processing step, multi-label text mining process may suffer from the problem of curse of dimensionality. Curse of dimensionality arises because of presence of huge amount of features, out of which many features may not be relevant for decision-making or may be redundant. Furthermore, the presence of irrelevant and redundant features complicates the mining process by generating ambiguous data representation and poorly describing category labels. Therefore, data cleaning phase aims at removal of such redundant features, and transformation phase tries to morph the original feature sets into new representation which will be efficient from storage and retrieval point of view. Feature extraction and analysis are commonly used in the data cleaning and transformation phase to unfold this challenge by reducing the original large feature space and retaining the relevant features. The process of feature analysis employed in typical pattern recognition system is depicted in Figure 6.2 and is carried out in two steps namely: parameter extraction and feature extraction. The information relevant for pattern classification is extracted from the input data in the form of a p-dimensional parameter vector X. In the feature extraction step, parameter vector X is transformed to a feature vector Y, which has a dimensionality m ($m < p$). The dimensionality of parameter vectors is normally very high and needs to be reduced for the sake of less computational cost and system complexity. In case of Big Data, there are enormous number of incrementally growing p-dimensional parameter vectors which necessitate the presence of feature extraction and analysis operation in transformation phase.

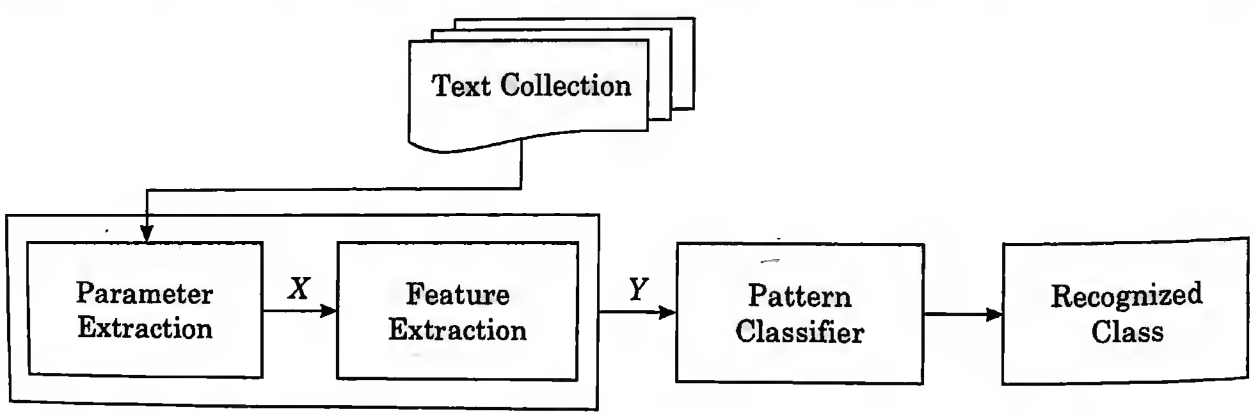


Figure 6.2 Feature analysis in pattern recognition system.

To be more specific, in the context of multi-label text mining, text collection serves as an input to parameter extraction phase. Tokenization, stop word removal, stemming and lemmatization and term weight calculation operations are carried out in parameter extraction phase. The filtered feature vectors serve as parameter X. It serves as the input to the feature extraction phase that transforms X and produces reduced set of feature Y. The reduced feature set is utilized by classifier training phase and based on it, class labels of test document are predicted.

There exists various well-known Feature Extraction (FE) techniques to extract the features from the input resources. This resource list includes text, image, protein synthesis data, etc. The traditional FE techniques are Principal Component Analysis, Linear Discriminant Analysis, Fisher Discriminant Analysis, Latent Semantic Indexing, Non-negative Matrix Factorization, etc. But it has been observed that most of the FE methods are not applicable to multi-label domain because of its associativity with multiple labels. And hence, we discuss some of them briefly here.

The dominant FE technique is PCA that transforms the data into a reduced space that captures most of the variance in the data. It uses the orthogonal transformation in order to convert a set of observations of possible correlated variables. The results of the PCA are usually discussed in terms of component or factor scores and loadings. However, PCA is an unsupervised technique in that it does not take class labels into account during transformation. PCA projects the data onto a single dimension that maximizes variance; however the two classes are not well separated in this dimension. By contrast, LDA strives for a transformation that maximizes between-class separation.

The goal of LDA is to separate the classes by projecting their samples from p-dimensional space onto a finely oriented line. Similar to LDA, FDA is also a well-known technique for reducing dimensions. This is done by maximizing the scatter between the classes while minimizing the scatter within each class, thereby, obtaining Fisher Optimal Discriminate vector. Finally, the projection vectors are computed using dot products of mapped samples. However, the transformation process becomes computationally intensive while extending to multi-label domain.

The next FE technique LSI is also based on unsupervised dimensionality reduction approach. For the application of LSI, the documents are first transformed in Vector Space Model (VSM) form. Thereafter, Singular Value Decomposition (SVD) is performed to find the sub-eigen space with large eigen values. Even then, LSI is not capable to incorporate any additional knowledge, which is more prevalent in multi-label setting. That is why subsequent years saw the rise of MLSI as the extension of LSI. The MLSI preserves the information of inputs, meanwhile capturing the correlations between the multiple outputs. The recovered latent semantics, thus incorporates the human-annotated category information, and can be used to substantially improve the prediction accuracy. But, MLSI ignores class discrimination information when applied to the whole training set.

In subsequent years, NMF has been introduced as an effective unsupervised FE technique for analyzing the latent structure of non-negative data such as images and documents. It imposes the non-negativity constraint in its bases and coefficients and provides a lower rank approximation formed by factors whose elements are also non-negative. NMF provides a more intuitive and meaningful decomposition allowing only additive operations. Moreover, NMF is successfully extendable to multi-label paradigm as well.

Data representation and modelling

The processed and transformed features need to be represented and modelled to facilitate efficient decision-making or predictions. There are many ways to represent text document such as Bag-Of-Words (BOW), N-gram representation, vector space model, graph-based model, tensor-based model, etc. In case of BOW, representation of every element in the vector indicates

presence or absence of a word in the document by binary or TF-IDF indexing. In this model, a text (such as a sentence or a document) is represented as an unordered collection of words, disregarding grammar and word ordering. This model is easy to implement and simple to understand. It retains the frequency of the words in the document, but looses the sequence of information. Some studies suggest that BOW representation outperforms many times because of its simplicity.

N-gram representation represents *n*-character slice of a longer character string. In case of multi-word string, word boundaries are identified by blank spaces. It retains word sequence information, and it does not require linguistic knowledge and offers a simple way of describing documents. It has been observed that, in this model, high dimensional vectors are generated. This requires huge storage space and computational complexity, leading to either over fitting or failure of powerful classification tools.

In case of vector space model (VSM), text is considered to be a bag of words. It is an algebraic model for representing text documents as vectors of identifiers. In vector space model, text document is represented as vectors. When VSM is used for large number of documents, vocabulary of terms is created and appearance frequency of term in document is used as the value of respective dimension in document vector. It is a unidirectional representation that means vector form can be created from document but document can not be regenerated from its vector form. However, the order in which the terms appear in the document is lost in the vector space representation. Tensor Space Model (TSM) based representation models the text by multi-linear algebraic high order tensor instead of the traditional vector. TSM is supported by the *High Order Singular Value Decomposition (HOSVD)* for dimensionality reduction and can identify latent structure of documents, thereby, improving classification performance.

Even though, all the aforesaid traditional data representation models are popular, they are not efficiently able to explore relationship between documents as well as labels, which is important in case of multi-label mining. Hence, for the multi-label applications where consideration of relationship matters, graph-based approaches are mostly preferred because of their ability to explore relationship. By considering the importance of graph based representation in multi-label context, we have explored them in detailed manner subsequently.

Exploratory data analysis and decision reporting

Big Data exploratory analysis involves mining relevant information from large volume of varied data in its raw form. In the context of multi-label data, there is a need to define a data and label model to emphasize what each element means in the context of the others. In the exploratory data analysis phase, traditional machine learning based algorithms are generally employed to analyze the data and report the results based on user request. Multi-label algorithms such as classifier chains method, pruned sets based method, multi-label decision tree, that is, C4.5, ML-*k*NN, ensemble-based methods, and etc. may be used for data analysis purpose. However, in order to extend their effectiveness for Big Data, certain issues need to be incorporated such as iteration capacity, adaptation to incrementally evolving labels, effective storage and retrieval management. In order to achieve this, generally the aforesaid multi-label algorithms are integrated with Big Data analysis tools such as BI tools, In-Database Analytics, Hadoop, Oracle advance analytics tool, etc.

◇ 6.3 GRAPH-BASED MODEL

This representation is useful for those applications where consideration of relationship may improve mining and decision-making performance. In this, the set of documents are represented in the form of a graph with document as a node or vertices and relation between them as a link or edge. In similar fashion, labels are also represented as nodes, and similarity between them represents the linking between them. Most of the time, the graph is weighted and cosine or kernel based similarity measures are used to calculate weight of an edge between two document. It expresses relationship between documents and their respective terms. However, graph structure poses important challenge of storage and retrieval speed which is more prevalent in case of Big Data analytics as data may be collected or stored at different places and that too in different formats. Hence, graph construction and its optimum representation from the efficient storage point of view are very important from Big Data point of view. In this view, graph construction phase is described subsequently.

◇ 6.3.1 Multi-label Graph Construction

A crucial step in graph-based representation is the graph construction by means of conversion of data into a weighted graph. The labelled and unlabelled text samples are served as vertices in a graph whereas, weighted edges between them are represented by the similarity score between the data sample pairs. In case of multi-label context, the small portion of labelled vertices is then utilized to predict the labels of unlabelled vertices by means of label prediction phase. It is observed that the graph construction method plays a key role in the performance of multi-label mining process. The basic steps employed in graph construction are depicted in Figure 6.3.

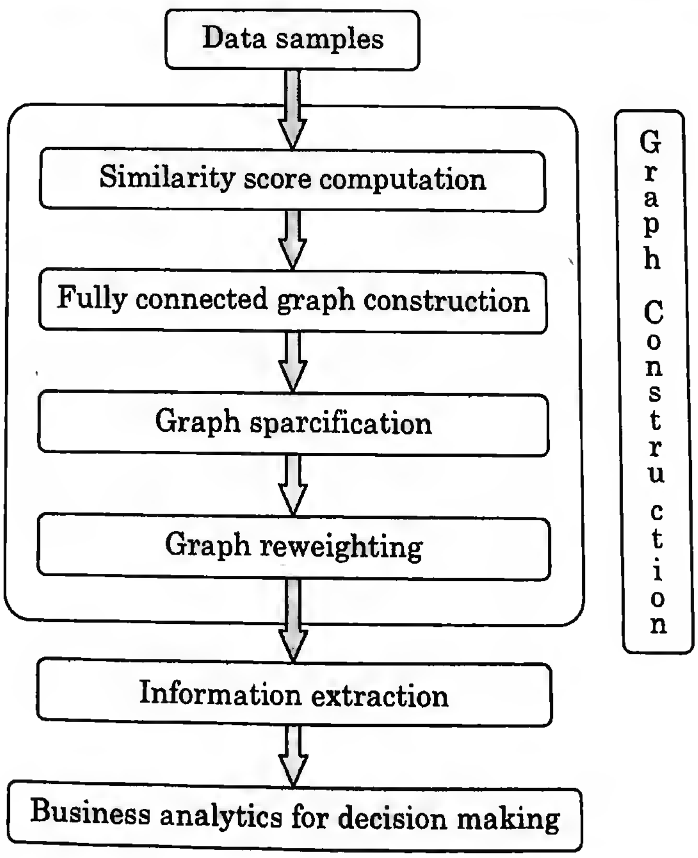


Figure 6.3 Basic steps in graph construction.

The labelled and unlabelled text samples are served as vertices in a graph. The pair-wise similarity score between all pair of vertices in the graph is then computed. Consequently, fully connected weighted graph is constructed. However, the storage and retrieval of fully connected weighted graph is computationally very expensive. In this view, graph sparcification step is applied subsequently. Graph sparcification plays a key role in graph construction and is responsible for sampling of vertices or edges in order to construct a new, smaller graph that is representative of the original graph resulting into improved efficiency in terms of storage and retrieval time. Graph sparcification may be divided into two types namely: neighbourhood-based methods such as, kNN, ϵ-neighbourhood and matching-based method like b-matching. The kNN graph connects k closest samples and ϵ-neighbourhood connects the samples within a distance of ϵ. Each vertex in b-matching graph has exactly b degree and generates more robust and balanced graphs.

Further, the edge reweighting process is carried out in order to produce final set of edge weights. It converts the unlabelled text data samples into a weighted sparse undirected graph in the form of adjacency matrix. Furthermore, the generated graph and label information are utilized by the subsequent information extraction phase in order to predict label set of unlabelled and test data. It is observed that sparcification is important since it leads to improved efficiency, better accuracy and robustness to noise in the decision-making stage.

Multi-label graph construction not only aims at generation of text graph but also gives emphasis on exploring label relations through label graph creation as depicted in Figure 6.4 and Figure 6.5 respectively. The process commences with creation of fully connected dense graph by computing the similarity score between each pair of feature vector. Further, the newly created document graph is sparcified and reweighted in order to improve time and storage requirement of the dense document graph. Thereafter, the fully connected weighted label graph is generated by computing similarity between each pair of label vector using suitable similarity measure, which is also sparcified and reweighted. Finally, the weighted and sparcified document and label graphs are utilized for further relevant information extraction and decision-making.

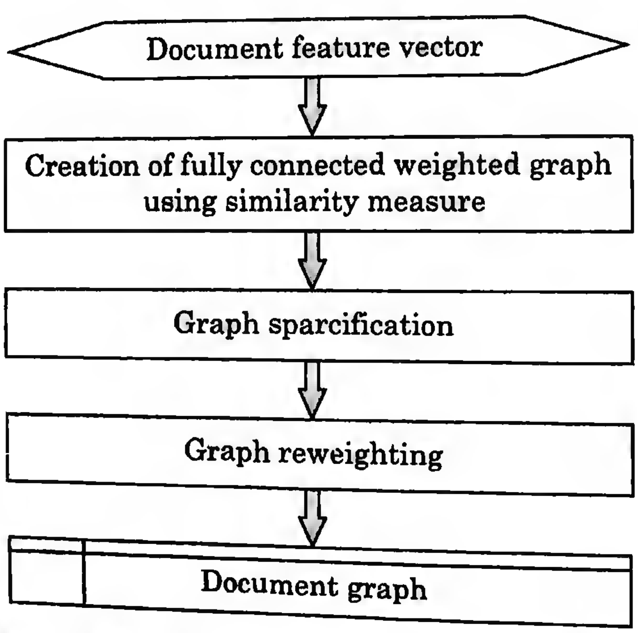


Figure 6.4 Basic flow of document graph creation.

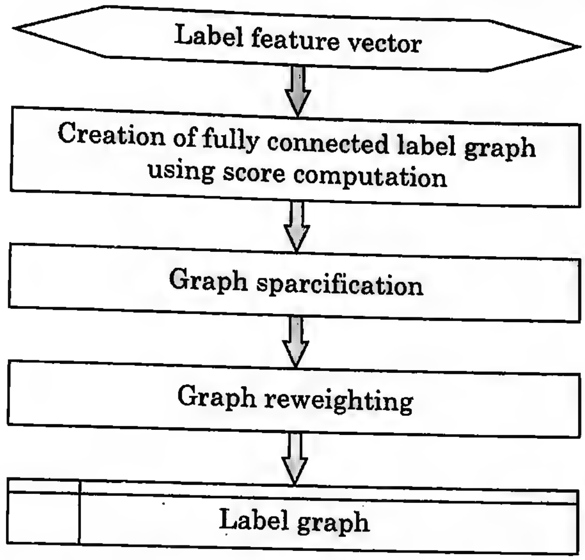


Figure 6.5 Basic flow of label graph creation.

By means of aforesaid strategy, the relationship information is preserved by means of document and label graphs in the process of multi-label text mining. But while extending it to Big Data environment, many factors need to be incorporated in order to choose graph representation method. The graph representation methods may vary depending on the application, nature of labels, size of documents as well as labels, etc. With this view, following subsection covers the traditional graph models and their application domains.

For example, In case of Banking system, millions of customer transactions are processed daily through traditional banking, mobile banking and internet banking. A unique customer ID may be associated with multiple labels such as saving account, housing loan account, car loan account, PPF account, etc. Here at abstract level, based on query generated the customer may belong to different set of labels. That is to say for fulfilling KYC norms, the single customer may be associated with multiple labels, whereas while processing credit details of housing loan account, the record of same customer may be searched with respect to housing loan account only which in this case may require to be processed at different branch. In this case, the label *housing loan* is in turn associated with multiple attributes such as Rate of Interest, Loan Amount, Tenure, etc.

The same scenario may be observed in case of analyzing sentiments of audience about the movie by means of reviews collected through social networking sites, blogs, comments from newspaper, spot feedback received, etc. The same movie may be associated with multiple labels such as entertaining, excellent, awesome, awesomely horrible, average, etc. All the reviews are contributing to generation of large number of textual information per day and within a fraction of second. While predicting some outcome based on these reviews, the relationship existing between intra clusters and inter cluster may be going to provide good insight into the prediction. The reviews gathered from teenagers about a movie may be different than that of from adults. These and many such scenarios not only emphasize multi-label mining but may generate more accurate predictions if relationship is explored properly. For this sake, we have described various representative graph-based Modelling and representation methods for relationship exploration in subsequent sections.

◇ 6.3.2 Traditional Graph-based Modelling Methods

Text document elements such as words, phrases, sentences and paragraphs are connected to another through various relationships. Relationship between elements is helpful to maintain overall meaning and discourse unity of the text contents. Many text document applications can be modelled using graph. Graph data structure is a strong representation of the text document in a way to show association between text elements and represents meaning and structure of a text document.

$$G = \{\text{Vertex, Edge relation}\}$$

$$\text{Vertex} = \{F, S, P, D, C\}$$

where, F = Feature term, S = Sentence, P = Paragraph, D = Document, and C = Concept

$$F = \{t_1, t_2, t_3, \ldots t_n\}$$

$$S = \sum_{i=0}^{n} t_i$$

$$P = \sum_{i=0}^{n} s_i$$

$$D = \sum_{i=0}^{n} p_i$$

$$DC = \sum_{i=0}^{n} d_i$$

Edge relation = {Structure, Syntax, Semantic}

Edge relation between two feature terms may different on the context of Graph.

1. Word occurrence together in a sentence or paragraph or section or document
2. Common words in a sentence or paragraph or section or document
3. Co-occurrence on the fixed window of n-words
4. Semantic relation: Words have similar meaning, words spelled same way but have different meaning, opposite words.

We explore study of Graph model into following two parts:

1. How graphs are built from text document.
2. What computations are done on text graph.

◇ 6.4 GRAPH REPRESENTATION

After pre-processing data samples, the features representing documents are taken into consideration for the construction of Graph. The location, order and proximity of term occurrence, which are discarded under the standard document vector representation models are preserved using graph model. The Graph construction is described for Web document and Text document.

◇ 6.4.1 Structural Representation of Web Document

A web page usually contains various contents such as navigation, decoration, interaction and contact information, which are not related to the topic of the web page. Furthermore, a web page often contains multiple topics that are not necessarily relevant to each other. Therefore, detecting the content structure of a web page could potentially improve the performance of web information retrieval.

Standard representation

There are three sections defined for standard representation title, link and text. Title contains the text related to the documents title and any provided keywords (metadata). Link is the anchor text that appears in hyperlinks on the document. Text comprises any of the visible text in the document (this includes hyperlinked text, but not the text in the documents title and keywords).

With this representation, the graph can capture structural information of text (location, relative location of words).

An example of a standard graph representation for a short English Web document having the title \SPORT NEWS", a link whose text reads \MORE NEWS", and text containing \ENGLAND FOOTBALL NEWS", is shown in Figure 6.6, where TL denotes the title section, L indicates a hyperlink, and TX stands for the visible text. There are five words occurred in the document: \SPORT", \NEWS", \MORE", \INDIAN", \CRICKET", which correspond to five nodes in the graph. Four edges in graph show the relations between words in the documents.

For instance, there is an edge from \SPORT" to \NEWS" labelled by \TI" meaning that \SPORT" immediately precedes \NEWS" in the title section.

Simple representation

No title or Meta data is examined and the edges in the graph are not labelled.

N-distance representation

Succeeding terms are connected with an edge that is labeled with the distance between them. Figure 6.6(a), (b) and (c) shows these three representations.

Absolute frequency representation

For nodes, this indicates how many times the associated terms appeared in the web document. For edges, this indicates the number of times the two connected terms are appeared adjacent to each other in the specified order.

Relative frequency representation

This representation is same as the absolute frequency representation, but with normalized frequency values associated with the nodes and edges.

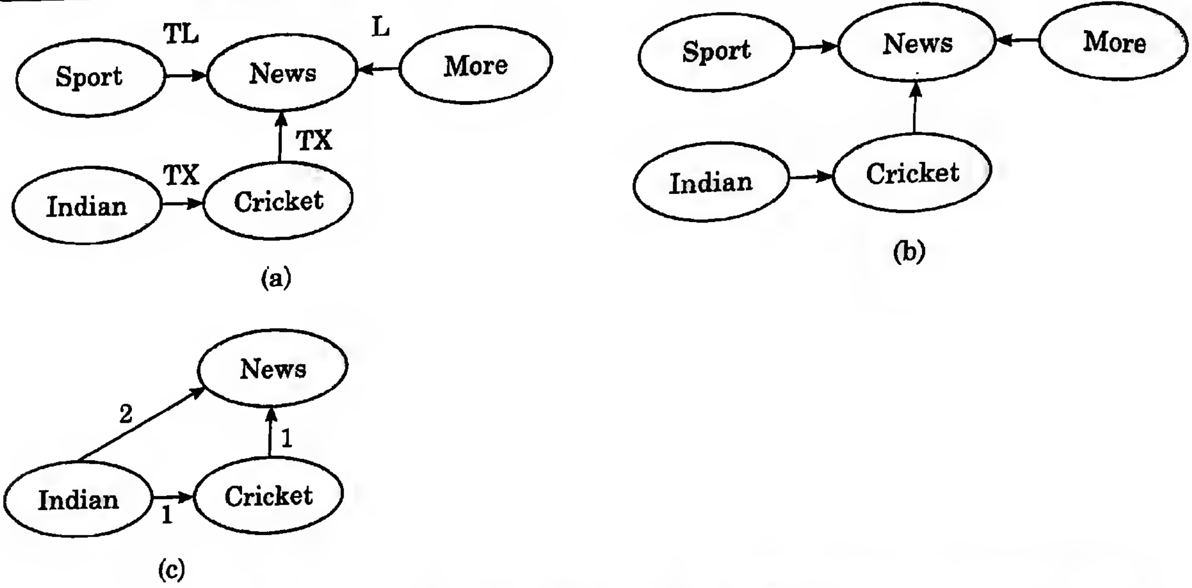


Figure 6.6(a) Standard representation (b) Simple representation (c) *N*-distance representation.

◇ 6.4.2 Structural Representation of Text Document

Pre-processed documents are considered for the representation. Each word is considered as a potential feature for a given term, all the terms that fall in the vicinity of this term are considered dependent terms. This is represented by a set of edges that connect the term to all the other terms in the window size generally considered of 2, 4, 6, and 8.

Sample text: Image processing is processing of images using mathematical operations by using any form of signal processing for which the input is an image, such as a photograph or video frame; the output of image processing may be either an image or a set of characteristics or parameters related to the image.

Structural representation of sample text is shown in Figure 6.7. This representation is successfully performed on a text classification task; the analysis achieves relative error rate reductions as compared to the traditional term frequency based approach.

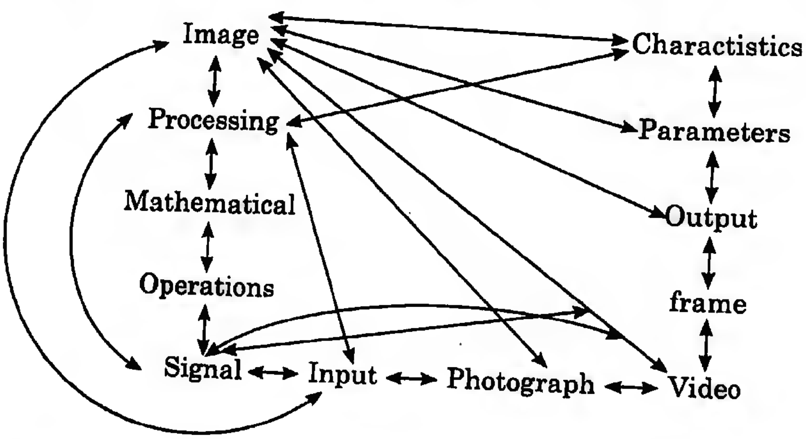


Figure 6.7 Sample occurrence graph drawn with window size 2.

◇ 6.4.3 Syntax-based Representation of Text Document

This representation uses syntax of term including part-of-speech tagging using graph that encodes word and tags dependencies. Part-of-speech tagging is defined as the task of automatically assigning parts of speech to words. Part-of-speech tagging is required by almost any text-processing task.

◇ 6.4.4 Semantic-based Representation of Text Document

Recently, there are many novel approaches other than using just words and relations between words. One of the methods is to capture semantic relations between words using conceptual graph. There are two nodes which are Concepts and Relation. Relation node indicates the semantic role of the incident concepts. For instance, the sentence 'John is wearing jeans' can be represented as a conceptual graph as shown in Figure 6.8.

Figure 6.8 Semantic representation.

Concept is shown by rectangles and Relation is shown as circles in the graph. John and Jeans play Agent and Object semantic roles in the current context.

◇ 6.4.5 Semantic Class

One of the major application for representing text as graph is semantic class construction. Semantic class construction is done by automatically extracting all elements belonging to certain semantic category (e.g. animals, fruits). Figure 6.9 illustrates a sample graph built to extract semantic classes.

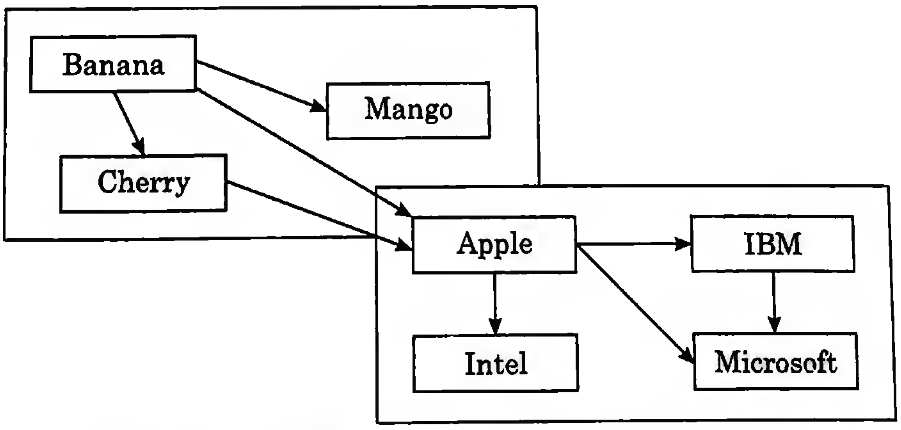


Figure 6.9 Semantic category example.

◇ 6.4.6 Semantic Network

Semantic network or Concept network (Figure 6.10) is a graph, where vertices represent

concepts and edges represent relations between concepts. The relations between concepts that are used in semantic networks are as follows:

- **Synonym:** Concept A expresses the same thing as Concept B
- **Antonym:** Concept A expresses the opposite of Concept B
- **Meronym, holonym:** Part-of and has-part relation between concepts
- **Hyponym, hypernym:** Inclusion of semantic range between concepts in both directions

Sample graph is displayed in Figure 6.10.

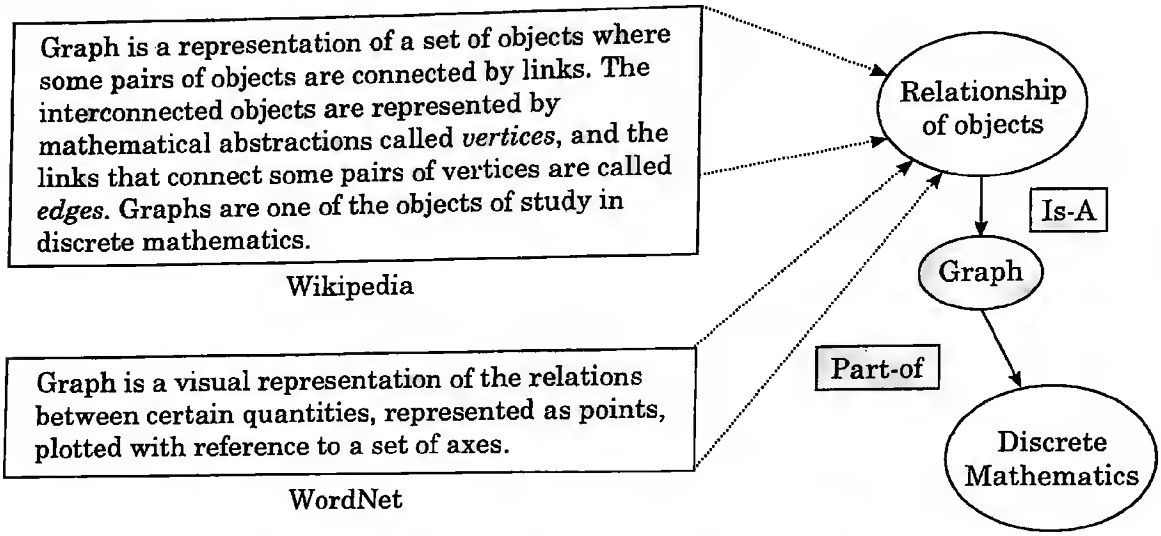


Figure 6.10 Semantic network example.

◇ 6.5 TEXT OPERATIONS USING GRAPH MODEL

Once Text document is modeled as graph, different graph methods can be applied to measure various properties of the graph and text document. This section gives overview of different graph methods applied to different text applications.

◇ 6.5.1 Sentence and Degree Centrality

The similarity between sentences is considered as a measure for association between sentences. Sentence centrality is used to cluster the sentences and for significant similarity degree centrality is used which helps to summarize a document.

$$\text{idf} - \text{modified} - \text{cosine}\,(x, y) = \frac{\sum_{w \in x,y} tf_{w,x} tf_{w,y} (\text{idf}_w)^2}{\sqrt{\sum_{xi \in x} (tf_{xi,x},\ \text{idf}_{xi})^2 \times \sum_{yi \in y} (tf_{yi,y},\ \text{idf}_{yi})^2}}$$

where $tf_{w,s}$ is the number of occurrences of the word in the sentences and is inverse document frequency.

◇ 6.5.2 Graph Topological Properties

Term co-occurrence in a document representing its association within n terms can be used as a relationship while constructing a graph. Graph topological properties like degree distribution, average path length, and clustering component help in ranking documents.

Average degree,

$$\partial(G) = 2\frac{E(G)}{V(G)}$$

Average path length is the ratio of number of vertices over its degree, i.e.,

$$l(G) \approx \frac{\ln(|V(G)|}{\ln(|\partial(G)|}$$

Clustering component of vertex v_i,

$$c(v_i) = \frac{2E(v_i)}{\partial(v_i)[\partial(v_i)-1]}$$

Average clustering component,

$$c(G) = \frac{\partial(G)}{|V(G)|}$$

where $\partial(G)$ denotes the average degree of graph $G.|E(G)|$ denotes the cardinality of edges in G and $|V(G)|$ denotes the cardinality of vertices in $G.E(v_i)$ is number of edges connecting the immediate neighbour of node v_i.

◇ 6.5.3 Local and Global Term Weight

Co-occurrence of term within sentence rather than n terms is considered as association. Degree centrality and closeness centrality are used to find local and global term weight of a term which is related to Term frequency and inverse document frequency. This concept is applied for Text classification and found better alternative to traditional TF – IDF factor.

$$IC - ICC_{t,d} = \frac{TC_{t,d}}{CC_t + 1}$$

where $TC_{t,d}$ is the centrality of a term in document d, and CC_t is the centrality of t in the graph constructed from the whole corpus.

◇ 6.5.4 Page Rank Surfer Model

A graph-based ranking algorithm implements the random surfer model where probability of jumping from given vertex to another random vertex in the graph is integrated.

$$S(v_i) = (1-d) + d * \sum_{j \in \ln(v_i)} \frac{1}{|\text{out}(v_j)|} S(v_j)$$

where ln (v_i) is pointing to its predecessor vertices and out (v_i) is pointing to its successor vertices and is a damping factor = 0.85.

This method is found suitable for sense disambiguation, where WordNet relations with other words in the sentence are used to find ranking of the senses. As an actual application for the problem of text classification, the results are encouraging.

◇ 6.5.5 Weighted Frequent Sub-graph Mining

Weighted frequent sub-graph mining (W-gSpan) is effective for selection of most significant construct for graph representation and this construct is used as an input for classification. Support count of graph G is support of G with respect to D,

$$\text{sup}(G) = \frac{\text{sco}(G)}{n}$$

Weighted support of G with respect to D is

$$W\text{sup }(G) = W(G) \times \text{sup }(G)$$

◇ 6.5.6 Graph-based Term Weight

Graph-based term weight by using different graph theoretic properties is described as follows:

TextRank: Higher the number of different words that a given word co-occurs with, higher the weight of these words, the higher the weight of this word.

TextLink: Higher the number of different words that a given word co-occurs with, the higher the weight of this word.

PosRank: Higher the number of different words that a given word co-occurs with and is grammatically related to, and the higher the weight of these words, the higher the weight of this word.

PosLink: Higher the number of different words that a given word co-occurs with and is grammatically related to, the higher the weight of this word.

These graph-based term weights are used for retrieval by integrated them into ranking function which ranks the documents with respect to queries. Table 6.1 shows different graph-based methods are applicable to different text operations.

Table 6.1 Graph-based Analysis Methods Used in Different Text Analysis Applications

Method	*Application*
Graph union	Document merging
Vertex ranking	Term/Sentence weight
Graph-based features like degree, clustering component	Text classification Text summarization Novelty detection
PageRank random surfer model	Semantic search
Sub-graph	Text classification Question–answer system
Graph matching	Plagiarism detection

◇ SUMMARY

Multi-label problem has received significant attention in the machine learning, information retrieval and NLP-based research so far. The varsity of issues in Big Data analytics like high dimensionality existing in the feature and label space, the increasing number of labels, association between label set collection and heterogeneity between textual collections for being collected from varied sources of text raises significant challenge to make multi-label methods suitable for Big Data analytics. This fact makes the task of multi-label Big Data mining even more complicated. This chapter highlights various solutions towards this challenge by discussing phases of multi-label mining: data collection, data processing, data cleaning and transformation, data representation and modelling, exploratory data analysis, validation and reporting decisions/ predictions. The chapter also emphasizes the need for appropriate text representation and importance of graph-based representation. Graph representation provides terms as vertices and relationship as edges. Relationship can be co-occurrence, grammatical, conceptual or semantic. Graph analysis methods like intersection, union and topological properties are effective for various text analytics for different applications. Various graph-based representation methods are elucidated along with case study and examples. This chapter provides a newer insight to researchers in the multi-label Big Data domain.

Multiple Choice Questions

1. Which of the following feature extraction technique maximizes the scatter between the classes while minimizing the scatter within each class?
   (a) PCA (b) LDA
   (c) FDA (d) MLSI
2. In which of the following scenario the property of association with more than one category makes the task of classifier more challenging?
   (a) Multi-label (b) Multi-class
   (c) Multi-instance (d) Single label
3. Which of the following phase in multi-label unstructured mining is responsible for removal of redundant features?
   (a) Data collection (b) Data representation and modelling
   (c) Data cleaning and transformation (d) None of these
4. Which of the following graphical model is constructed by automatically extracting all elements belonging to certain semantic category?
   (a) Co-occurrence graph (b) Concept graph
   (c) Syntactic graph (d) Semantic graph
5. Which of the following characteristics of text document is used to calculate graph-based term weight?
   (a) TF (b) TF/IDF
   (c) N-gram (d) Threshold

Concept Review Questions

1. Perform text document representation using semantic network.
2. Represent text document as co-occurrence graph and perform ranking using graph topological properties like degree distribution and clustering component.
3. Represent text document as co-occurrence graph and perform ranking using page rank surfer model.
4. Represent various phases of unstructured multi-label mining for the online shopping cart system.
5. For the automated news classification system, design the multi-label text classifier by defining necessary pre-processing. Explain the labels to be used, and specify graph model required to model relationship between labels with justification.

Critical Thinking Questions

1. Explain the phases in multi-label unstructured data mining for the application of the GPS-based automated vehicle tracking system.
2. Design a graph model for text document and use graph algorithms for text summarization.

Laboratory Assignments

1. Implement association of text elements in document using graph model.

   [**Hint:** Use term weight and feature extraction.]
2. Write a program to construct label graph for domain data. Construct co-occurrence graph for the profile data. Use appropriate similarity method to compute the association between profile and domain text data for the application of industry recruitment system.

7

Distributed High Dimensional Data Clustering for Big Data

—Dr. Sunita Jahirabadkar

◇ 7.1 INTRODUCTION

Data mining deals with the problem of extracting interesting patterns from the data by paying careful attention to issues like computing, communication and human–computer interaction. Clustering is one of the primary data mining tasks which aims at dividing datasets into subsets or clusters in such a way that the objects in one subset are similar to each other with respect to a given similarity measure (Figure 7.1).

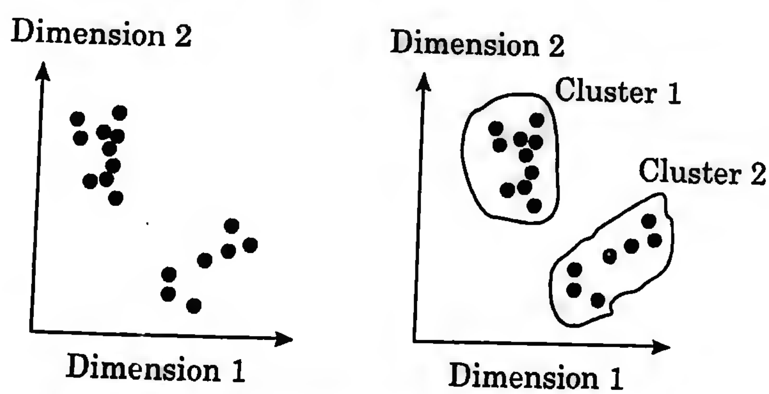


Figure 7.1 Clustering of data-based on the distance between data objects.

Objects belonging to different clusters may not remain close to each other with respect to similarity measure being used. Clustering algorithms help to understand the natural groups within a dataset. Clustering as a data mining task has many applications in the areas such as business intelligence, image processing, medical science, geology, environmental science and

so on. Clustering also works as data compression technique or as a pre-processing step for many other data mining algorithms, e.g., as in classification to create labelled, training data.

Even though, huge amount of work is already being reported in the field of clustering, there is still huge need of new approaches to deal with the recent capabilities of generating huge amount of unstructured, distributed data. Clustering such real world datasets needs to deal with Big Data.

Conventional commercial data mining systems are designed to work as centralized vertical application, on top of Data Warehouse—like architecture. But in real world, data is distributed among several sites, and each site generates its own data and manages its own repository. The transmission of the entire local data set is often unacceptable because of privacy and security aspects and bandwidth constraints. Analyzing and mining these distributed sources require distributed data mining techniques. Distributed data mining is expected to perform partial analysis of data at individual sites and then to send the outcome as partial result to other site where it is sometime aggregated to the global result.

Typical clustering methods compute similarities between objects based on an entire set of selected attributes. However, when the number of measured attributes is large, it may be the case that two given groups differ at only a subset of the measured attributes, and so only a subset of the attributes are 'relevant' to the clustering. In such cases, traditional clustering methods may fail because the differences between any two groups, averaged over all the attributes, are small. Subspace Clustering algorithms are clustering algorithms that look for and build clusters not necessarily in the whole space, but also in subspaces of the attributes.

◇ 7.2 APPLICATIONS OF DISTRIBUTED SUBSPACE CLUSTERING

Many of the real world distributed datasets consist of objects modelled by high dimensional data. Each object is described by hundreds of attributes. For instance, in many computer vision applications, such as motion segmentation, face clustering with varying illumination, pattern classification, temporal video segmentation etc., image data is huge-dimensional and distributed. Other examples for high-dimensional feature vectors representing distributed complex objects can be found in the area of molecular biology, CAD database and text databases.

The following application areas will express the need for Distributed Subspace Clustering approach for mining high dimensional distributed data.

◇ 7.2.1 Financial Data Analysis

With the evolution of information technology and increase of economic globalization, financial data are being generated and collected at an exceptional speed. As a result, there has been a crucial need for automated approaches to effective and efficient utilization of massive amount of financial data to support companies and individuals in strategic planning and investment decision-making. Data mining techniques have been used to uncover hidden patterns and predict future trends and behaviours in financial markets. The competitive advantages achieved

by data mining include increased revenue, reduced cost and much improved market place responsiveness and awareness. With the globalization, the data is spread all over the world. Financial data is no exception to this. Hence, distributed clustering algorithms plays major role in finding general properties of financial data. For example, customers with similar behaviours regarding banking and loan payments may be grouped together by multidimensional clustering techniques.

In finance and sales, to identify the different subspace clusters that exist in the huge amount of sales data, we can find which of the different attributes are related. This can be useful in promoting the sales and in planning the inventory levels of different products. Thus, effective distributed Subspace Clustering methods can help to identify customer groups, frauds or unusual transactions and facilitate targeted marketing.

◇ 7.2.2 Biomedical and DNA Data Analysis

In last few years, there has been a tremendous research in biomedical. A great work has been done in the study of human genome by discovering large-scale sequencing patterns and gene functions. DNA analysis discovers the genetic causes of many diseases and disabilities. It also helps to discover the new medicines and approaches for disease diagnosis, prevention and treatment.

As such, all DNA sequences are comprised four basic building blocks, called nucleotides. These nucleotides are combined to form long sequences or chains that resemble a twisted ladder. Human beings have around 1,00,000 genes. Each gene is comprised of hundreds of individual nucleotides arranged in a particular order. Thus there are nearly unlimited numbers of ways that the nucleotides can be ordered and sequenced to form distinct genes. It is challenging to identify particular gene sequence pattern that plays roles in various diseases.

Since many interesting sequential pattern analysis and similarity search techniques have been developed in data mining, data mining has become a powerful tool in the problems like definition of the molecular variability of a population of bacteria, or finding the groups of co-expressing genes. We never know beforehand if we are going to find one unique group of homogeneous individuals or many groups, and we do not have an idea of how many individuals per group we are going to find. These problems are approached by clustering methods.

But due to the highly distributed, uncontrolled generation and use of a wide variety of DNA data, distributed clustering techniques can play an important role in semantic integration of such data. Further, it is well-known in molecular biology that only a small subset of the genes participates in any cellular process of interest and that cellular process takes place only in a subset of the samples. Furthermore, a single gene may participate in multiple pathways that may or may not be coactive under all conditions, so that a gene can participate in multiple clusters or in none at all. A 'block' is a sub-matrix defined by a subset of genes on a subset of samples. Thus, to capture coherence exhibited by the 'blocks' within gene expression matrices, subspace clustering methods have to be used.

Text data is by default high dimensional data. Big Data is by default distributed data. Thus, clustering the unstructured Big Data which is distributed across multiple locations, the best approach is distributed subspace clustering. This chapter, thus, describes in detail these clustering approaches specific to unstructured data and Big Data.

◇ 7.3 HIGH DIMENSIONAL DATA CLUSTERING

High dimensional data clustering is the cluster analysis of data having few tens to few hundreds of dimensions. Here, the data is made up of a set of objects, described with a big collection of features, known as feature vector.

Classic examples of high dimensional data can be found in the areas of satellite image processing, pattern recognition, text data mining, CAD (Computer Aided Design) databases, bioinformatics, information integration systems, etc. High dimensional data fetches an exceptional attention and requires to take extra efforts as compared to conventional clustering algorithms. In particular, the traditional clustering algorithms fail in the cases of high dimensional data due to the inherent sparsity and obviously do not produce meaningful clusters.

There are three main challenges in high dimensional data clustering.

◇ 7.3.1 Curse of Dimensionality

Different clustering algorithms use different ways of similarity measures (or distance measures) to compute the closeness between various data objects. There are many similarity measures available in data mining field such as distance based, pattern based, density based, etc. As such, different measures result in different clustering models. However, in distance-based cluster analysis, the distance between two objects is considered to indicate similarity or dissimilarity between two data objects (Figure 7.1). Euclidean distance measure is the most commonly used distance measure technique which computes the distance between any two data objects by calculating the differences between the values of attributes.

The traditional way to measure the distance between any two data objects is by calculating the distance between these objects along each dimension and then using any of the standard distance formula such as Euclidian distance or Manhattan distance, etc. However, in case of high dimensional data, measuring distance using this traditional way, faces the problem, historically known as 'Curse of dimensionality'. It means, data objects become sparser and sparser as the number of dimensions increases. As the number of dimensions increases, the distance between any two data objects becomes uniform and thus, the sparsity increases exponentially. In such scenarios, the distance-based clustering algorithms fail. Hence, such distance-based clustering algorithms may not prove as useful while clustering high dimensional data.

The concept of Curse of Dimensionality is illustrated in Figure 7.2 using 200 data objects, which are generated randomly, against which maximum and minimum distance difference among every pair of data objects is drawn.

For certain data distributions, as the number of dimensions increases, relative differences in the distances between the closest and the farthest data objects tend to zero. Therefore,

$$\lim_{d \to \infty} \frac{\text{MaxDist} - \text{MinDist}}{\text{MinDist}} \to 0$$

here, d denotes number of dimensions.

Above equation defines the potential problems in clustering high dimensional data, where the internal data distribution generates uniform distances among various data objects.

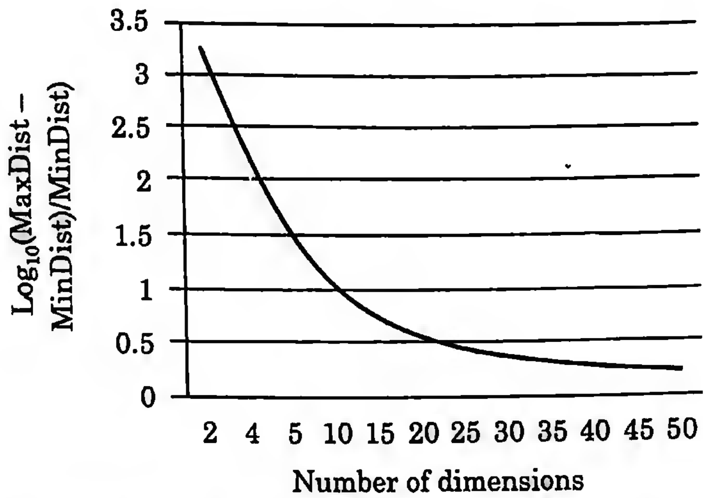


Figure 7.2 Curse of dimensionality.

◇ 7.3.2 Irrelevant Dimensions

Another major difficulty in high dimensional data clustering is that many dimensions are irrelevant from the clustering perspective. Clustering on these dimensions may not create meaningful clusters. For example, if we cluster students' database using their email ids, as the email ids are unique, this may not distribute students in proper groups. Thus, these irrelevant dimensions confuse the clustering algorithms by generating noisy clusters. The most commonly and traditionally used solution to solve this problem is to reduce the dimensionality of data, without losing the meaningful information from the given database. Feature selection is the most oftenly used approach before actually applying clustering, which aims at removing irrelevant dimensions from the given data.

However, in high dimensional data, clusters can be found in various subsets of dimensions. For the purpose of clustering, one particular dimension may be useful in forming one dimension-combination, whereas, it may be irrelevant in some other combinations. Thus, a global filtering approach for feature selection is not feasible.

◇ 7.3.3 Correlations among Dimensions

In case of high dimensional data, there may be large number of attributes and some correlations among them. So, it may be possible that the clusters are not parallel to axis, but are arbitrarily oriented.

The quality of any clustering algorithm is highly dependent on the number of dimensions as well as specific dimensions that are used for the clustering process.

Thus, there are two major approaches to handle the problem of high dimensional data. In the first approach, variety of dimensionality reduction techniques can be applied prior to clustering, to reduce the dimensionality of the given dataset. In such a case, after reducing the dimensionality, any existing traditional clustering algorithm can be applied on the database. The other way would be 'subspace clustering'. In high dimensional data, clusters are embedded in various subsets of the entire dimension space. A new research area of high dimensional data clustering, known as 'subspace clustering' detects such clusters embedded in various subspaces.

◇ 7.4 DIMENSIONALITY REDUCTION

Dimensionality reduction techniques help to reduce the number of dimensions from the given high dimensional data.

Fundamental approaches to remove irrelevant attributes of the data are 'Feature Transformation' or 'Feature Selection' approach.

Feature transformation

Feature transformation approaches project the higher dimensional data onto a smaller dimensional space. The only care needs to be taken in this method is to preserve the distance among the original data objects. These approaches apply dimensionality reduction, aggregation techniques, etc. to summarize data as well as to create linear combinations of the dimensions. These kinds of techniques are effective in analyzing the data in few cases, as these can effectively reduce the noise. The most popular approaches of this kind are Principal Component Analysis (PCA), Singular Value Decomposition (SVD).

The major limitation of feature transformation methods is that these methods do not eliminate any of the dimensions. They just transform high dimensional data into its linear combination. This makes them retain irrelevant dimensions (or not-so-useful dimensions) while transforming high dimensional data to low dimensional data, which makes the clusters less meaningful. Hence, such types of feature transformation methods are best suited in databases, where there are no irrelevant dimensions.

Feature selection

As compared to feature transformation methods, feature selection methods attempt to eliminate a few of the irrelevant dimensions from the given high dimensional data. Feature selection approaches search through different subsets of the attributes and evaluate these attribute subsets for clustering. However, the major limitation of these techniques is that they translate many dimensions into one set of dimensions. This makes it difficult later, to interpret the clustering results.

Many feature transformation and feature selection approaches are available in the literature, to reduce the dimensionality of high dimensional text data improving the quality of text data representation, making it more appropriate for clustering text data.

Further, when clusters are hidden in various subsets of the high dimensional data, these approaches become inappropriate. The most commonly used approach, which is the extension of feature selection process, is subspace clustering. Subspace clustering looks for the clusters hidden in various subsets of the high dimensional feature space of the data. Subspace clustering algorithms, thus, first search for relevant subsets of dimensions in the complete feature space and then the clusters hidden in those subsets of the dimension space.

◇ 7.5 SUBSPACE CLUSTERING

Subspace clustering is an evolving methodology which, instead of finding clusters in the entire

feature space, aims at finding clusters in various overlapping or non-overlapping subspaces of the high dimensional dataset. They find large number of applications in the area of image processing, computer vision, CAD databases, text data mining, information integration system and so on.

Formally, a subspace cluster C in database DB is defined as, $C = (S, O)$, where $O \subseteq DB$, $S \subseteq A$ and S is a subspace of dimensions in attribute set A.

Figure 7.3 shows the subspace clusters, Cluster 1 to Cluster 5. Cluster 4 represents a traditional full dimensional cluster ranging over dimensions d_1 to d_{16}. Cluster 3 and Cluster 5 are non-overlapping subspace clusters appearing in dimensions $\{d_5, d_6, d_7\}$ and $\{d_{13}, d_{14}, d_{15}\}$ respectively. Cluster 1 and Cluster 2 represent overlapping subspace clusters as they share a common object p_7 and a common dimension d_6.

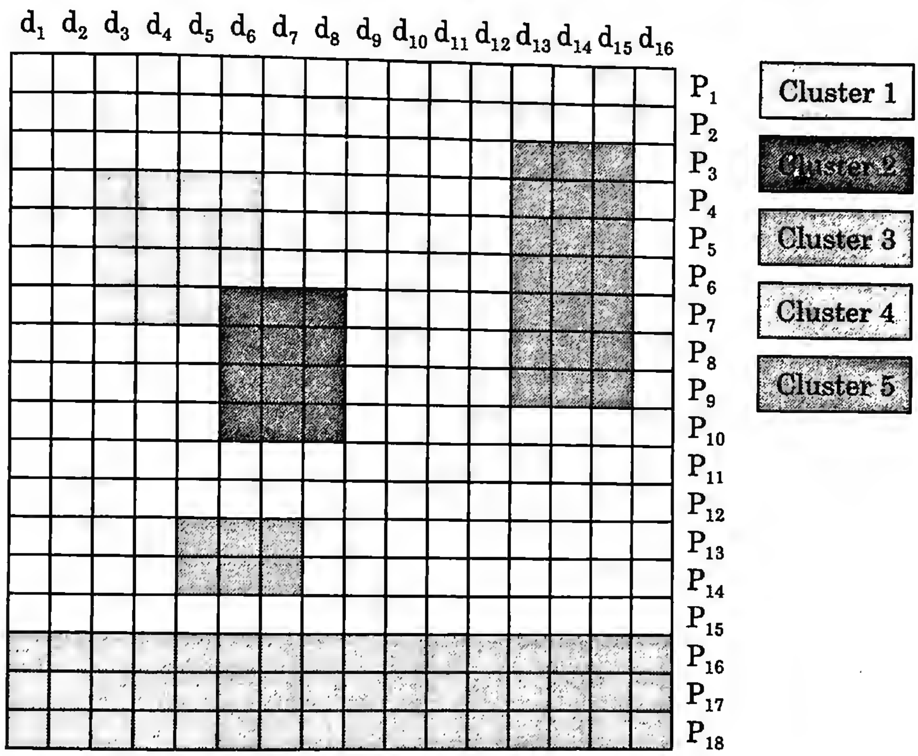


Figure 7.3 Overlapping/non-overlapping subspace clusters.

Subspace clustering algorithms face two major challenges. Initially, searching for the relevant subsets of dimension space, which encloses quality clusters. Once the relevant subspaces are located, it needs to discover clusters in each of these subspaces.

However, the search space for relevant subspaces is infinite. This makes it essential to apply some heuristic approach to make the subspace searching process feasible. The heuristic approach applied to restrict the searching of relevant dimension sets, determines the characteristics of the subspace clustering algorithm. Once the relevant subspaces of the high dimensional space are identified, any suitable clustering algorithm can be applied to explore the hidden clusters in that subspace.

Like any other clustering algorithm, subspace clustering algorithms should be efficient and produce high quality, interpretable clusters. These algorithms must be scalable with respect to the number of objects as well as with the number of dimensions.

The first subspace clustering algorithm, CLIQUE was proposed by R. Agrawal. Later, lots of noteworthy algorithms have been proposed in data mining literature. While all these

algorithms classify data objects into various groups of data objects or clusters, each of them uses different method to define clusters. These algorithms make various assumptions for input parameters. The clusters are defined as fixed size or varying size, overlapping or non-overlapping clusters and so on.

Later, many significant algorithms have been presented in literature. While all these approaches organize data objects into groups, each of them uses different methodologies to define clusters. They make different assumptions for input parameters. They define clusters in dissimilar ways as overlapping or non-overlapping, fixed size and shape or varying size and shape and so on. The choice of a search technique, such as top–down or bottom–up, can also determine the characteristics of the clustering approach.

P. Lance, et al., suggested in a well-known survey, the two major classes of subspace clustering algorithms using the searching strategy, as top down subspace clustering approaches and bottom up subspace clustering approaches. Ilango, et al., classified high dimensional clustering approaches as partitioning approaches, hierarchical approaches, density-based approaches, grid-based approaches and model-based approaches and further presented a survey of various grid-based approaches. S. Karlton, et al., classified subspace clustering approaches into two categories, density-based clustering and projected clustering. H.P. Kriegel, et al., classified different high dimensional data clustering approaches as subspace clustering (or axis parallel clustering), correlation clustering (or arbitrarily oriented clustering) and pattern-based clustering.

Many significant subspace clustering algorithms exist in the data mining world, each having different characteristics caused by the use of different techniques, assumptions, heuristics used, etc. A comprehensive classification scheme needs to be defined which will classify existing approaches into various appropriate classes.

Clustering or grouping text documents into a conceptually meaningful groups or clusters is a significant application of high dimensional data clustering. In this, a collection of unstructured documents are represented using a set of significant or important context-bearing words from the document, called vector space model or bag-of-words model. These words then form the feature space of the text documents. Typically, even a small document includes large number of words or features, making the document vector a very high dimensional. Further, if we see the collection of such documents in text database, a single document contains a small number of total bag-of-words. Hence, the document vector for each word is quite sparse. And thus, we need to understand meaningful features from the vector space to apply clustering on these documents. Subspace clustering can play a major role in such cases to select meaningful subspace from the feature space of text data.

◇ 7.6 DISTRIBUTED SYSTEMS

Traditional commercial data mining systems are meant to work as centralized vertical application on top of data warehouse like architecture. However, in many organizations data is distributed among several, independently working locations which are connected to each other through LANs (Local Area Networks), WANs (Wide Area Networks), etc. The example organizations can be supermarket chains like IKEA, COOP, etc. international business houses having branches spread across the globe like Microsoft, Volvo, Ericsson, etc. The transmission of the entire local

data set is often unacceptable because of privacy as well as security aspects and bandwidth constraints. In some application areas, transmitting entire data to the central location is almost impossible, e.g. astronomy, satellite data, etc. Analyzing and mining these distributed data sources require distributed data mining techniques.

Distributed data mining is expected to perform partial analysis of data at individual sites and then to send the partial results as the outcomes to a central site where they are aggregated as a global result. Figure 7.4(a) shows a traditional centralized architecture where data from various sources is collected at a central warehouse and data mining tools are applied to get the interesting patterns. Whereas, Figure 7.4(b) shows distributed clustering which combines clustering with communication. In Figure 7.4(b), on each local site, individual data is analyzed to carry out independent clustering, and a local model is created. This local model, holding partially aggregated data is sent to a central site. The central site then analyzes these local models arriving from different sites, to create the final, global clustering model.

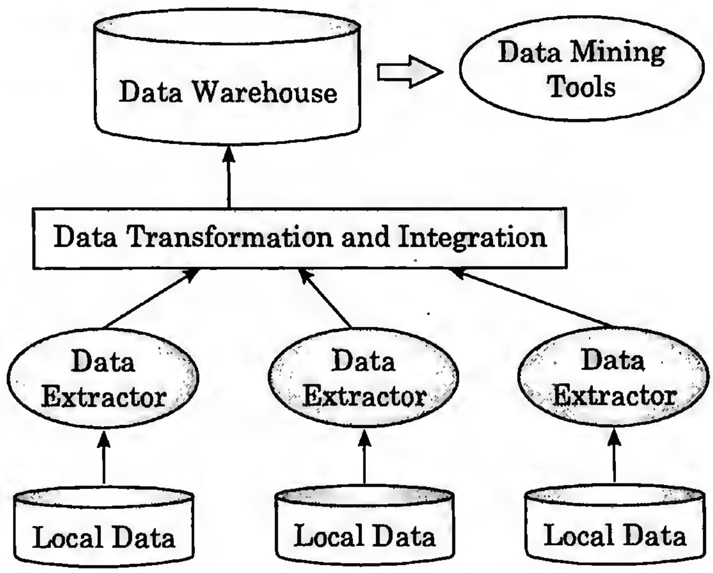


Figure 7.4(a) Traditional centralized architecture for clustering.

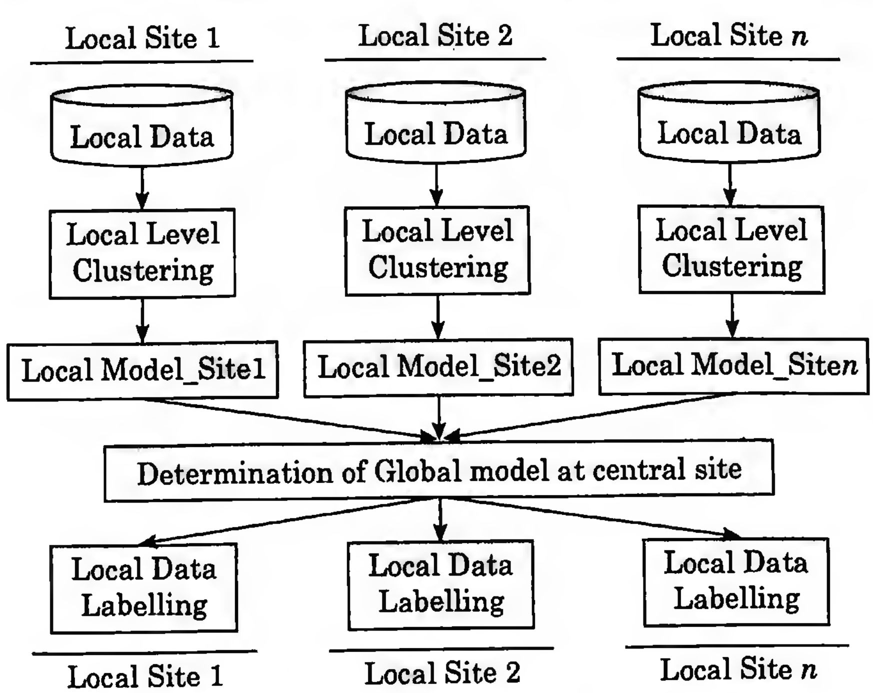


Figure 7.4(b) Distributed clustering.

The results thus generated at the central site, may be sent back to each of the local sites to mark the local data into a global context.

This requirement of extracting knowledge from distributed data without collecting it to a central site, defines a new research area called DKDD (Distributed Knowledge Discovery in Databases).

For implementing distributed clustering algorithms, there are lot of facets which required to be considered such as the type of data on which clustering needs to be applied (such as text data, web data, genes data, etc.), type of distributed data (such as homogeneous data or heterogeneous data), the type of environment in which clustering needs to be run (such as P2P, computer clusters, LAN, WAN, etc.), criterions (such as privacy preservation, bandwidth requirement) and so on. All such information is very essential to design, implement and evaluate the distributed clustering algorithm.

In distributed database systems, data may be stored in multiple computers located over a dispersed set of locations connected through an interconnection network (Figure 7.5). A distributed database system possesses loosely coupled database sites which do not share physical components.

In a distributed system, a database administrator can distribute chunks of data from a given database, across multiple physical locations. A distributed database can be located on intranet, extranet or network servers on the internet.

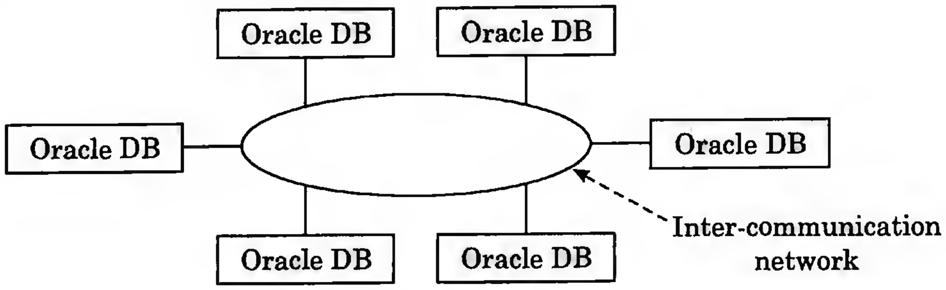


Figure 7.5 A simple distributed database architecture.

◇ 7.7 TYPES OF DISTRIBUTED DATABASES

The distributed databases are mainly categorized as homogeneous and heterogeneous databases.

Homogeneous distributed database

If the distributed database comprises identical hardware as well as software on all locations, and may appear as if a single database, then it is called 'homogeneous distributed database'. A homogeneous database system is simple to design and manage. It needs to hold the following conditions at each location:

- The operating system must be same and compatible
- The data structures used must be same and compatible
- The database management system or database application used must be same and compatible

Heterogeneous distributed database

If the database comprises varying hardware, software, database management systems or even data models, then it is called 'Heterogeneous distributed database'. It may adopt different schema as well as software. For example, one location may use traditional file processing systems or old database management system software, whereas, another may have most modern database management technology to store the data. One location may work on Windows environment, while the other may work on Linux.

This makes heterogeneous databases quite complicated as the query processing and transaction processing face major problems. In heterogeneous systems, individual local sites request the database access using their local query language. A translation system converts these commands to allow communication between various sites. Heterogeneous system is not often feasible from technology or financial point of view.

◊ 7.8 TYPES OF TRANSMISSION OF DATA

There can be three different ways of transmitting data among distributed data sources throughout the clustering process. They are:

(i) **Whole datasets:** This is the most simple and straightforward way of communication in which the peer local sites exchange their complete data needed for the clustering algorithm. However, it does not then satisfy any of the above mentioned constraints, making it most inefficient way of clustering.

(ii) **Representative data:** In this, a few of the data objects as the representatives of the cluster are transmitted to the central site. The most suitable representatives are those objects which can correctly represent the cluster. It satisfies the bandwidth and communication cost constraints, however, it does not satisfy privacy constraint as the actual data objects are transmitted.

(iii) **Cluster prototypes:** At each local site, each cluster is represented using cluster prototype such as centroid, dendrogram, etc., and this prototype is transmitted to the central site. This satisfies all the above constraints and hence the most popular way of transmitting data.

◊ 7.9 ADVANTAGES OF DISTRIBUTED DATABASE SYSTEMS

Compared to parallel systems, distributed database systems have many advantages, such as:

- It increases availability, efficiency, reliability and accessibility of the database.
- It provides modularity by allowing adding or removing sites from the distributed database system, without affecting overall system.
- It can be built using the local sites which are independent of location, hardware, operating systems, software, database management systems, network, etc.
- It allows local sites to control their own data providing them local autonomy or site autonomy.

- In catastrophic circumstances such as fire, etc., distributed data saves major part of the database as it is not stored at only one place, but distributed across multiple locations
- It is economic to create a network of small computers, than to use a single computer with enormous power.

◇ 7.10 DISTRIBUTED CLUSTERING

Many of the distributed clustering algorithms are straightforwardly derived from the algorithms which were earlier developed for parallel clustering. These algorithms assume that a single database is divided into multiple locations and hence the database is homogeneous. However, in distributed environment, extra design work needs to be made to take care of the diverse nature of the database.

As compared to distributed data mining systems, distributed clustering process involves design decisions based on the criterions as what needs to be achieved such as accuracy, privacy, communication cost, bandwidth, etc. and how the clustered data has to be analyzed. If privacy preservation is of utmost priority, then algorithms which send the actual cluster data at the central location may not solve the problem. Or, if network bandwidth preservation is the criterion, then size of the resultant local models to be sent to central location may matter a lot. These design decisions eventually decide the nature and characteristics of distributed clustering algorithm.

Steps of the standard distributed clustering algorithm can be summarized as follows:

(i) A local cluster model is generated at each local site.
(ii) These local models from different locations are collected at a central site.
(iii) A global cluster model is generated using all the local models and the final clustering information is sent back to corresponding local sites to mark the clustered data.

Each local site performs clustering operation independent of each other. Thus, taken out of the distributed context, each local site can apply a traditional, classical clustering algorithm for local clustering. In case of high dimensional data clustering, a local clustering can be preceded by feature selection process or feature reduction process to reduce the dimensionality of the data. However, use of subspace clustering to select the smaller dimensional, but relevant subspaces for discovering meaningful clusters at each local site, is the most innovative contribution of our research work.

Aggregation of the local models at a central site depends upon local clustering techniques used at each local site. For example, if the local models are generated using the partitioning notion of clustering or grid-based clustering, then the global model also needs to be based on partitioning technique or grid-based technique.

◇ 7.11 TEXT DATA CLUSTERING

Text mining algorithms process text data which is unstructured in nature, to mine or extract significant information or pattern from the text and make it available for further statistical, machine learning or data mining algorithms such as clustering or classification, etc. The

information derived from the text data is based on the words contained in the documents or documents containing specific keywords. Thus, we can analyze the keywords appeared in various documents, cluster the words or documents to determine similarities between them, compare the keywords that how they are related to other documents available on World Wide Web and so on.

Typical applications of text data mining includes analyzing various market surveys, automatic processing of emails, messages, documents, etc., automatic classification of texts, emails, etc. to identify junk mails or to automatically route messages to appropriate departments and so on. Another type of applications of text data mining can be found in analyzing contents of text documents such as analyzing insurance claims, warranty documents, diagnostic prescriptions, competitor's websites, etc.

Thus, text mining can be considered simply as a process of converting a text document into a numerical representation. As a simplest method, all the words discovered in a set of input documents will be counted, in each document, and a matrix kind of data structure will be maintained to store frequencies of each word occurred in each document. Certain common words such as 'a', 'an', 'the', 'or', 'and', etc. (called stop words) are excluded from this matrix to make the list more meaningful and less complex. Further, a process called 'Stemming' is applied on the words to understand the basic form of the word and combine different grammatical forms of the same words together. For example 'counting', 'counted', 'count', etc. will be combined to form one single entry in the matrix. Once a set of documents is represented as a matrix of unique words/terms along with their frequencies of occurrence, various standard, well-known data mining or machine learning analytical techniques can be applied on this matrix. These techniques may include efficient information retrieval of documents, clustering, classification, predictive data mining and so on.

Clustering can be found as most useful technique in text data mining domain. Clustering divides a collection of documents into various categories or groups in such a way that group of documents in one category portray one topic or context such as photography, music, health, entertainment or Indian history and so on. Text data clustering has many applications such as grouping web documents, grouping web search results and so on.

For clustering, the text data objects can be of different granularities such as words (or terms), sentences, paragraphs or the whole documents. The major application of text data clustering is in information retrieval to organize documents to improve retrieval and support browsing. Organizing the documents hierarchically to form logical categories, helps to improve browsing of a collection of documents, for example, scatter/gather technique which allows efficient and systematic browsing using clustered organization of documents. Another application of text data clustering is in corpus summarization in which a logical summary of the collection of words is formed which is used further to provide insight into the underlying corpus. Sentence clustering can also be used for document summarization. Document classification, a supervised variant of clustering, can also be used to improve the quality of document clustering algorithms.

Clustering, being an unsupervised learning has to face many challenges. However, clustering unstructured data faces few more challenges. The most importantly, volume of text data is too huge. Dimensionality of text data is another major challenge. However, many of these dimensions are not useful in clustering, making the task more complex. Text clustering

algorithms need to face these challenges to be scalable at high volume of data and efficient even for high dimensional data. And moreover, need to handle data semantics and data sparsity.

Text data sources are mainly unstructured such as web documents containing text, images or other multimedia data; or unstructured such as XML data. However, existing text clustering algorithms are based on structured data. Thus, to apply clustering on text data, the original unstructured data needs to be transformed into structured format. The commonly used structured format to represent text documents is Vector Space Model. This simplest representation of text data represents each document in the form of matrix of unique words/terms along with their frequencies of occurrence. In transformation of original text data to the Vector Space Model, a number of pre-processing steps are used, including filtering, stemming, term frequency calculation, term selection, etc. These pre-processing steps are very important because they could significantly affect the results of text clustering.

Many general purpose clustering algorithms such as *k*-means clustering algorithm or other general purpose hierarchical/partitioning algorithms can be directly applied on this representation to achieve text data clustering. An improved representation of text data is based on weighing methods such as TF-IDF weighting (Term Frequency–Inverse Document Frequency) which includes assigning weights to each word or term based on the frequencies of the individual words in the document as well as frequencies of words in an entire collection of documents. Many general purpose clustering algorithms which are based on quantitative data, such as it can be used on this representation to determine the most relevant groups of words in the text data.

Similar to general purpose clustering algorithms, text clustering algorithms are also classified as partitioning clustering algorithms, hierarchical clustering algorithms, and parametric Modelling based methods, for example EM algorithm.

◇ 7.12 DATA REPRESENTATION FOR CLUSTERING TEXT DATA

Even though the simplest representation of text data is in the form of matrix of unique words and their frequencies, the text data has many distinctive properties which require specialized algorithms to be designed to perform the data mining tasks. These exclusive characteristics of text data representation are as follows:

- Each uniquely identified word in a text document forms a dimension for that text data object. Thus, the dimensionality of the text representation is very large. However, there cannot be always close relationship/distance among the terms and hence the underlying data is sparse. If the text object is very short such as paragraphs or simple sentences or tweets, then this problem becomes even more serious.
- However, when the words are typically correlated with one another, the feature space will be still large; however the principal component or the core concept of the document will be much smaller. This also needs to design specialized algorithm for text data clustering.
- In a given set of documents, each document may have different number of words. Thus, it becomes important to normalize the document representations appropriately during the clustering task.

Thus, the high dimensional representation of the text data or documents demands the design of text-specific algorithms for document representation and processing, and same with clustering.

The TF-IDF representation of document (also called 'Vector space model') normalizes the word frequencies (TF) with their frequency of appearance over the whole set of documents (IDF). This normalization of word frequencies helps in clustering text documents, as it reduces the weight of each such term which has occurred more frequently in a document. This helps in trimming down the importance of terms which are very frequent in one document and raises the importance of those terms which are more discriminative but with less frequency across the set of documents. Further, often to avoid detrimental effect of any single, very high frequent term, a sub-linear transformation is applied to the term frequencies of words in a document. Document normalization itself is a very wide area of research. Many other techniques of normalization are available in literature, as in.

◇ 7.13 TEXT CLUSTERING SYSTEM

Many general purpose clustering algorithms such as *k*-means clustering algorithm or other general purpose hierarchical/partitioning algorithms can be directly applied on the text data by representing it into structured representation. This requires a pre-processing step before applying clustering on the set of documents. The major challenge in clustering large number of unstructured text data is to understand as well as interpret the results of clustering applied on the vector space model of documents. It is still possible to interpret clustering output by looking into the contents of documents, if the number of documents is small. But, if the number of documents is large, it is not feasible to read every document to read contents of all documents. Thus, we need to extract a few keywords from each of the cluster to understand the semantic of the documents grouped in that cluster. This requires post-processing step in clustering text data.

A text data clustering thus needs a complete system which will first convert unstructured, huge amount of varied data into a structured, compact data format; apply clustering and then interpret/utilize the results to understand meaning of the clustered documents. Such a text data clustering system mainly consists of five modules.

Figure 7.6 shows these five modules, followed by brief description of each module.

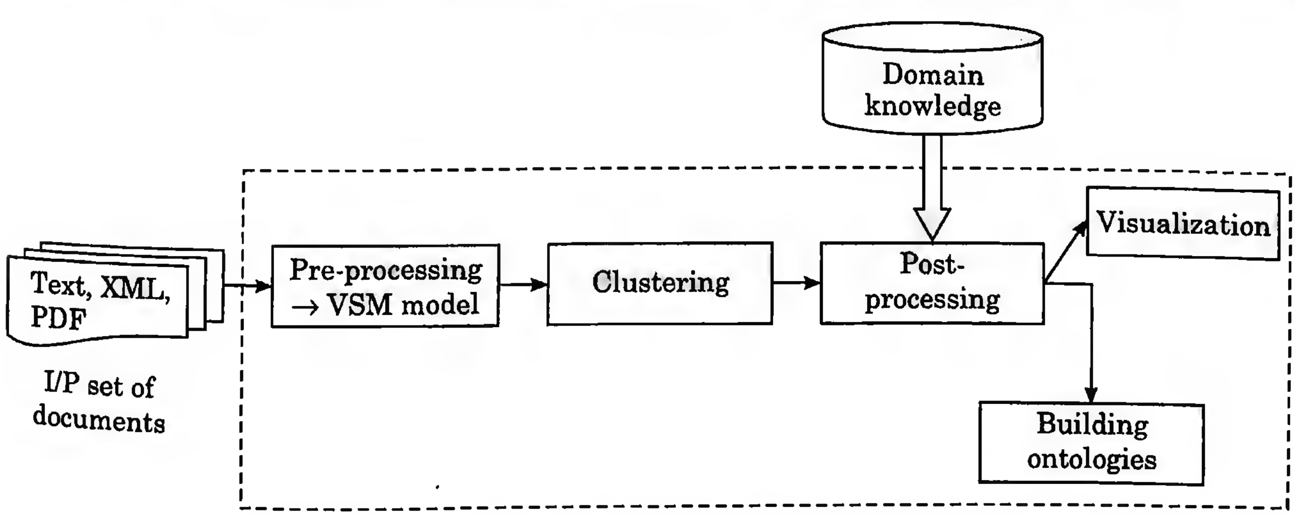


Figure 7.6 Text data clustering system.

Pre-processing

This module includes functionalities to transform a real time unstructured data into a structured representation in Vector Space Model, so that any clustering algorithm can be directly applied on it. The main functionalities include parsing (to convert document into a set of words by removing spaces and punctuation marks), stop word removal (to remove not-so-useful words such as a, an, the, is, was, as, has, etc.), stemming (to convert different forms of words into one original form such as singing, sing, sang into one word-sing), understanding synonyms, homonyms, etc. and term selection and term weighing such as TF-IDF. There are a few tools available in market which allow to perform this pre-processing step.

Clustering

This module can use any type of clustering algorithm on the set of documents, which are represented as Vector Space Model in the pre-processing module. It searches for similar type of documents based on the words forming context of underlying document set and represents each cluster with identified topic. This step can be further extended to apply subspace clustering algorithm, so that the significant features of the document can be identified making high dimensional text data clustering more efficient and effective process.

Post processing

This module uses an electronic database to mine a few (generally 4 to 10) representative words from each of the cluster representing topic of the cluster. Further, it also searches for common keywords among various clusters to derive/define relationship between different clusters.

Visualization

This module helps to visualize the semantic of each cluster using their keywords or significant words in each cluster. It also portrays the relationship among various document clusters using the common keywords falling in each cluster document. There can be many methods to represent document clusters. One method can be to represent each cluster as a semantic network. Each node can represent a document cluster which can be described with a small set of significant words. The edges between two nodes can represent the relationship between two clusters. Other than this, many other visualization tools can be configured to visualize document clusters in various graphical formats.

Building ontologies

Building ontologies module helps to build ontologies for each set of document using the clusters formed in earlier step. These ontologies help to interpret and understand the specific domain of text documents.

The next section details the subspace clustering approach for text data clustering.

◇ 7.14 SUBSPACE CLUSTERING IN TEXT DATA

The quality of any clustering algorithm is highly dependent on the number of dimensions as well

as specific dimensions that are used for the clustering process. In text clustering, each significant term forms a dimension. However, commonly used terms such as *in*, *or* do not contribute to clustering efficiency, rather increases the complexity. Thus, it becomes most vital decision to effectively select the dimensions for clustering, so that to reduce the dimensionality of text data and to reduce the noisy words in the corpus which may harm the clustering efficiency. Nowadays, the concept of 'ontology' is used to characterize the domain of the text document. Ontology of a text document represents the document's semantic, hierarchical conceptual model to understand the context of a document along with relation between various words within that document. However, ontologies for documents are manually created by domain experts by understanding the key context, important words and relationship among them in each of the document. A major amount of research work is taking place in this area to automate this process of generating ontologies.

Subspace clustering can play a significant role in automating the process of generating ontologies by learning and understanding the key domain of the document. Subspace clustering algorithms, as the first step, search for the relevant subspaces in the feature space and then, they find clusters in each of these subspaces. In case of text data, each document is represented as a vector, including set of words enclosed in the corresponding set of documents. An individual document usually contains a small fraction of the entire number of words. Thus, the document vector may contain many zeros at the place of words, not part of that document. Subspace clustering algorithms can play a role here to find subspaces, i.e., to understand relevant keywords/features from the large vector representing important context of the text document. Further, subspace clustering algorithms will also play a role of feature reduction technique. If the set of documents are represented as document term matrix, in which each row or instance represents one text document and each feature represents the keywords in the document, the subspace clustering algorithm will generate a set of relevant keywords (features/subspaces) for the corresponding set of documents. These keywords build the main context of corresponding group of documents. The second part of subspace clustering algorithms is to search for clusters enclosed in each of the relevant subspaces. Eventually, the clusters found in this clustering step represent the domain of the document along with important keywords represented by the subspace. This information of documents can be utilized further by many data mining applications such as to building classifier model in various classification algorithms. This classification model can be used then to classify large number of web pages according to their domain making information retrieval more time efficient. For example, subspace clustering can identify domain of each document as medical, health, finance, music, sports, entertainment related, etc. and clustering algorithm can group documents or web pages according to each domain. The classifier, trained using subspace clustering algorithm, can then classify and label newly added document in the existing document set.

The standard subspace clustering algorithms such as SUBCLU, PROCLUS or ORCLUS are based on the standard partitioning clustering algorithms. Another hierarchical subspace clustering algorithm HARP was presented in recent times. It automatically selects relevant dimensions for each document cluster. Standard *k*-means clustering algorithm which is most popular for clustering large amount of data, can be effectively modified to use it efficiently in large text data clustering.

◇ 7.15 BIG DATA CLUSTERING

With the great increase in use of social networking sites such as Facebook, Twitter, LinkedIn, etc., there is a spectacular growth in the generation and storage of general purpose data. It then becomes a major challenge to use this large-scale data, also called 'Big Data', for various information retrieval applications or many other data mining applications, such as clustering, classification, prediction, etc. Clustering, being an unsupervised functionality, can be used to find hidden patterns from this enormous amount of data.

However, there are two key challenges in terms of computation in clustering Big Data. Big Data has inherently heterogeneous features, as the data is being collected from various sources and different feature construction methods are used to store this data at various sources. For example, in an university database system, various linked colleges can use different representation schemes to represent students, their personal as well as other details, their test results marking scheme and so on. In biological data store, each human gene can be measured and represented using various representation techniques such as Single Nucleotide Polymorphism (SNP), gene expression or array comparative genomic hybridization, etc. In image data store, each image can be described by various descriptors such as SIFT, HOG, LBP, etc. Each type of the feature in the given data can represent specific information in the given data.

Thus, Big Data being heterogeneous data, the major challenge is how to integrate this Big Data which is spread across many nodes in distributed environment? And further, on which features to integrate this heterogeneous data? Another challenge is, how to cut down the computational cost required to perform clustering on large scale data?

The traditional *k*-means clustering algorithm is based on the centroid of the cluster. It partitions the database into a various clusters based on the distance of each object to the centroid of the cluster. Being simple to understand and implement, requires less computational cost even for large amount of dataset, *k*-means algorithm can be commonly used to cluster large scale distributed Big Data. However, *k*-means algorithm is mainly designed for single view data clustering application. We need a robust but less complex clustering algorithm similar to *k*-means, to integrate various features of Big Data. Thus, a Big Data clustering algorithm should satisfy the following characteristics:

1. It should be easily parallelized on multi-core powerful processors, to cluster big, distributed data.
2. It should be robust to noise as well as data outliers.
3. It should be able to produce more steady results even in different data initializations.

◇ SUMMARY

Within the process of KDD (Knowledge Discovery in Databases), data mining defines various functionalities applied on the database to discover interesting patterns and trends from the large data. Clustering is the fundamental and one of the vital data mining tasks. The methods and concepts presented in this chapter contribute to the field of distributed subspace clustering of

unstructured, high dimensional, distributed and Big Data. In this chapter, prior to providing Big Data clustering, various preliminaries related to unstructured, high dimensional text data clustering and distributed clustering are presented.

This chapter highlights various solutions towards this challenge by discussing phases of multi-label mining: data collection, data processing, data cleaning and transformation, data representation and modelling, exploratory data analysis, validation and reporting decisions/predictions.

This chapter highlights various challenges involved in high dimensional data clustering such as curse of dimensionality, irrelevant dimensions and correlations among various dimensions. It also discusses the issues related to existing dimensionality reduction techniques to handle high dimensional data for clustering. There are many application areas where this type of data mining methodology must be very useful. Thus, with respect to few applications, it further illustrates these challenges and other facets of distributed high dimensional data clustering. Text data clustering, which is a special case of high dimensional data clustering, is detailed in later sections along with discussions on converting plain unstructured text data to structured vector form and the whole text data clustering. It also emphasizes that the subspace clustering is the best applicable methodology for clustering text data as well as to Big Data. Thus, this chapter provides a newer insight to researchers in the Big Data clustering.

Multiple Choice Questions

1. Which of the following data mining methodology is used in data compression?
   (a) Data preprocessing (b) Clustering
   (c) Classification (d) Outlier analysis
2. Traditional clustering algorithms fail in case of high dimensional data and do not produce meaningful clusters due to:
   (a) Large datasets
   (b) Inherent sparsity of data
   (c) Data is present in the form of documents
   (d) Incorrect selection of distance measure
3. In a heterogeneous distributed systems, each location has:
   (a) Same and compatible operating system
   (b) Same and compatible data structures
   (c) Different schema as well as software
   (d) Same and compatible database management system
4. .... is not an attribute selection method.
   (a) Attribute construction (b) Decision tree induction
   (c) Subspace clustering (d) Feature selection
5. The commonly used structured format to represent text documents is:
   (a) Cosine similarity model (b) Stemming model
   (c) Vector Space Model (d) Document normalization model

Concept Review Questions

1. Describe and elaborate following terms: clustering, distributed clustering, high dimensional data clustering, subspace clustering.
2. Explain the concept of 'curse of dimensionality' with respect of challenges involved in clustering high dimensional data.
3. Elaborate the difference between centralized data mining systems and distributed data mining system.
4. Explain with suitable diagram text clustering system along with importance of pre-processing methods in clustering text data.
5. Write a short note on subspace clustering in text data.

Critical Thinking Questions

1. Use DNA synthetic data available on web and implement any feature selection/feature transformation and subspace clustering algorithm to understand how these algorithms select dimensions/attributes to find natural grouping or clusters hidden in the database. Comment on the best method for attribute selection for high dimensional data such as DNA data.
2. An email database is distributed across multiple sites. A typical distributed clustering methodology needs to be applied, in which local data at each site will be clustered locally and only cluster representative data will be sent to the global site. A global clustering model will be built at the central site. Suggest the best possible clustering algorithm which will handle distributed huge data and will take care of unstructured nature of email data.

Laboratory Assignments

1. **Application:** Customer segmentation according to their interests.

   Aim: Clustering of customer profiles to derive various interest groups of customers. The customers can have multiple interests. So, the groups will be overlapping.

   Data objects: Customer profiles

   Output: Subsets of attributes (defining interest areas) and clusters (defining customers belonging to each group)

   Challenge: Deriving relevant attributes and then applying clustering in each attribute subset.

   Problem statement

   Create or use synthetic high dimensional data (number of dimensions ranging from 30 to 50) to represent customer profiles. Minimum number of data objects – 100. Use SUBCLU, a subspace clustering algorithm in Weka and find out relevant subspaces for the given dataset (consider those subspaces which involve interest attributes. Minimum subspace dimensionality = 3). Apply *k*-means clustering algorithm in each subspace

to find customer groups. Compare the results of clustering with any other clustering algorithm such as DBSCAN, EM, Cobweb, etc.

2. **Application:** Context-based document clustering.

Aim: Clustering text documents to find groups of documents based on similar topic/context.

Data objects: Text documents described by their contents using Vector Space Model.

Output: Detecting groups of documents according to different topics/context.

Challenge: Understanding topic/context of each document and then grouping documents in various clusters.

Problem statement

Create or use synthetic high dimensional text data, represented in the form of vector space model. Apply *k*-means clustering to compute similarities in text document groups based on closeness factor of document vectors.

8

Machine Learning and Incremental Learning with Big Data

—Dr. Prachi Joshi

◇ 8.1 INTRODUCTION

Over the years, it has been observed that there is a tremendous demand of analysis of the data—the data that is growing at a large pace. Business Intelligence (BI) and analytics are all what is heard everywhere. To have this, it is Machine Learning (ML) that takes the lead.

Machine learning is all about identifying, predicting or forecasting. The way we humans learn, it is necessary that the analysis of the data takes place in an incremental way. The discovery of the patterns, the predictions, classifications and clustering is the task performed by machine learning algorithms. Machine learning approaches are capable to solve critical applications and differ to the traditional statistical analysis. They capture the trends, the changes and are in position to predict the drifts.

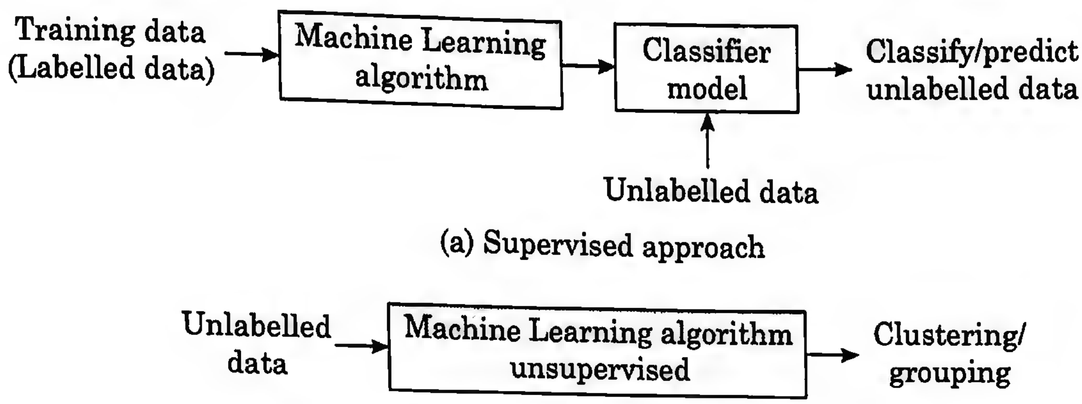
Machine learning approaches have seen enormous applications from weather forecasting, text classifications and many more. At present, there are many factors that have made Machine Learning a potential contributor in the Big Data domain. The issues of processing and analyzing this data are needed to be taken care of. Machine Learning techniques today find a place in this Big Data analytics. To mention a few examples, they are used in recommender systems, stock analysis, predictive systems and many more.

Let us explore more about Machine Learning, Big Data along with incremental learning.

◇ 8.2 MACHINE LEARNING: CONCEPTS

Typically, the machine learning methods are classified into supervised, unsupervised and semi-

supervised. The supervised methods work with labelled data—data with known classes. The job of a supervised approach is to classify an unknown sample. For this, it has to be trained with training data that is labelled. The approach builds a model based on this training. This is further used for classifying the unknown one. In case of unsupervised, it works with unlabelled data. The data is essentially used to form clusters or groups based on the similarities that are encountered in them. Whereas, in case of semi-supervised learning, it works in combination of the both, i.e., labelled as well as unlabelled data. Figure 8.1 depicts the approaches.



(a) Supervised approach

(b) Unsupervised approach

Figure 8.1 Supervised and unsupervised approach.

Let us look at few of the other aspects in machine learning paradigm.

Adaptive learning

There are two aspects in adaptive learning. One, where adaptive machine learning approaches enable to perform selection of an appropriate algorithm for a given problem, and the other, where the adaptive approach can be applied to update the inferences. This needs to happen with respect to the changes occurring over a period of time. The approaches face more challenges when real time response is expected and the environment is continuously changing.

Multi-perspective learning

Learning based on limited information and single perspective is not sufficient and might not be accurate as well. What seems to be accurate from one perspective could be misleading. Multi-perspective based learning aims at combining the relevant aspects from all the perspectives based on the scenario and the problem. The learning methods further need to prioritize the information available from the different perspectives so that the decision taken by the classifier is not biased. This is very important when a series of decisions are to be taken, and failure to capture the essential perspectives could result in incorrect decisions and affecting the overall performance.

Deep learning

It deals with multiple levels of representations and abstractions, in order to sense the different data. The entire intention of Deep Learning is to bring the Machine Learning closer to Artificial Intelligence. Typically, let us say for representation of images, Deep Learning decomposes

them and separates them out in different parts using multiple layers to determine the identity of that image. This learning works in an incremental way. Traditional Machine Learning approaches are observed to be shallow in nature, but Deep Learning expresses the output after it goes through a series of non-linearities. It is based on the principle of layered approach. The learning paradigm is motivated on intuition and neuroscience. At present, the most popular Deep Learning models are the Convolutional Neural Nets which are used in recognition tasks.

Active learning

Labelled data is often hard to collect. This collection is time consuming and needs expert to explicitly label them. Traditionally, passive learning approaches rely on the availability of the entire labelled data for the learning phase. In case of Active Learning, the learner actively chooses the data that is required to be labelled. There are different approaches to perform the same. One is query-based approach wherein outcomes of queries are used to get the data labelled. There could be selective approach as well working on it, where on availability of new data, the learning model decides to trigger a query or not for the data.

◇ 8.3 BIG DATA AND MACHINE LEARNING

In this section, we are shifting our focus on role of Machine Learning approaches with Big Data.

At present, we hear everyone talking about Big Data—the buzzword. Currently, Big Data analytics is the aspect that is looked at. Extraction of meaningful information from this large collection of data is a challenging task. Though many statistical approaches exist to perform this analysis, they rely on static analysis that might yield incorrect results. Moreover, the data is changing. The 3Vs—velocity, volume and variety—of the data also need to be taken care of. Machine learning approaches are powerful tools that can address this. They are often considered for predictive analysis in business domain. In this section, we would highlight the necessity of capturing and performing analysis of the Big Data with Machine Learning.

Till this chapter, though you must have come across many examples of Big Data like retail banking, sports activity, social media or any other, in all the cases, the amount of data accumulated is tremendous. Machine Learning approaches assist in providing an insight into the transaction thereby extracting the patterns and trends, and provide appropriate predictive analysis.

When we talk about Machine Learning approaches with Big Data, the need of the time is efficient processing of the real-time data at a faster speed. The Machine Learning approaches also have to consider the continuous stream of data and should be capable enough to handle the same.

We are already familiar with variety of Machine Learning techniques available for this purpose like association mining, pattern matching with different similarity based measures, classification and clustering as well. More precisely, Bayesian approach, Support Vector Machines (SVM), ensemble methods, decision trees (supervised learning methods) are found to be most common among them. Figure 8.2 explains the tasks of Machine Learning in Big Data.

To make use of Machine Learning techniques in Big Data, the most common available framework is Hadoop that has extensive library of Machine Learning Algorithms, called Mahout.

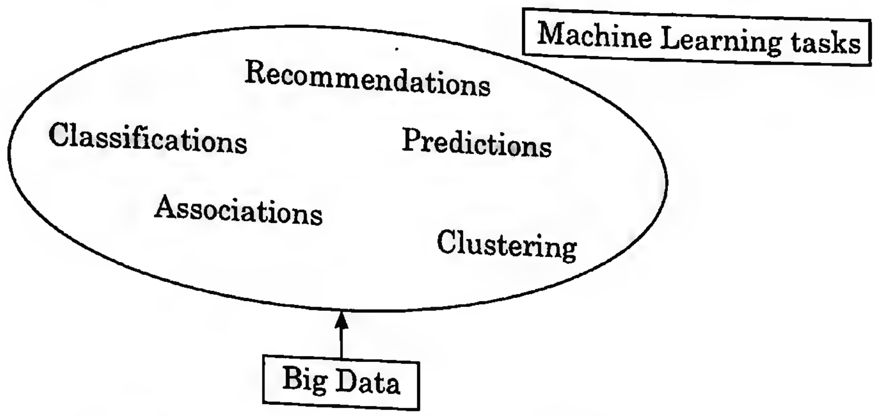


Figure 8.2 Machine learning tasks.

◇ 8.3.1 Mahout

Mahout, the library of the machine learning algorithms, is used for the various tasks. One point to mention is Mahout should not necessarily be used with Hadoop. But generally since Hadoop deals with Big Data and Mahout is required for some sort of recommendations, they go hand in hand. Figure 8.3 depicts simplified Mahout internal architecture.

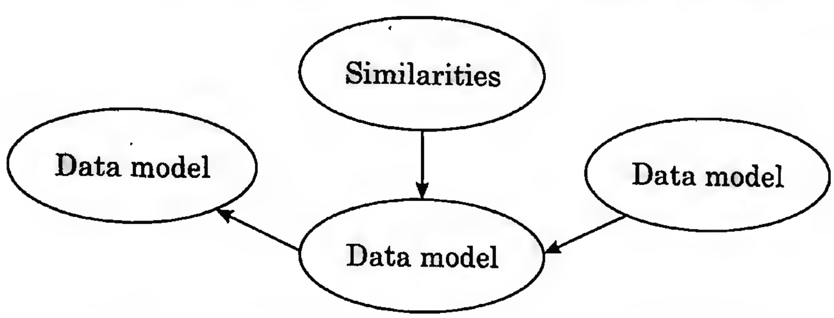


Figure 8.3 Simplified mahout architecture.

Assume that a simple recommender system is to be built. Such a system needs to consider the previous user preferences and the choices. In Mahout, from the database or the data store or KB that we refer to, the previous required data is been saved and the architecture performs recommendation say based on some similarity aspects or any other techniques.

Let us now explore more about incremental learning with Big Data.

◇ 8.4 WHAT IS INCREMENTAL LEARNING?

Intelligence is an inherent characteristic in mining. Business Intelligence is all about having this mining activity to justify and propose analysis in terms of predictions, forecasting or classifications. Machine learning is the underlying approach that deals with this intelligent mining. Among the traditional approaches of machine learning that are pre-dominant, viz. supervised, semi-supervised and unsupervised learning, which perform this task, often face challenge while learning from the new data that is generated over a period of time.

Owing to substantial growth in the data over a period of time and the necessity of analysis of the data in real time has given rise to concept of 'Incremental Learning' (IL). Incremental

learning is a learning paradigm that can accommodate newly evolved data, preserve the previously learnt facts and provide decisions based on it. The learning paradigm is capable of being adaptive to the environmental changes and possesses the ability to be selective in the learning process. Figure 8.4 depicts the incremental learning paradigm.

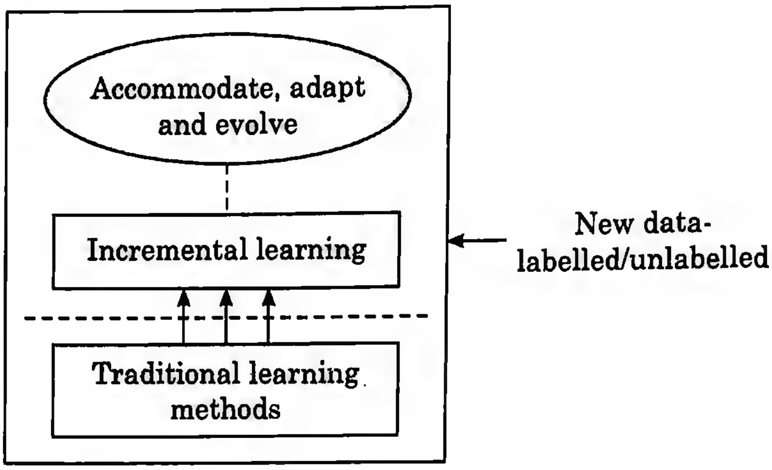


Figure 8.4 Incremental learning.

Incremental Learning approach thus needs to possess three basic properties:

Accommodate: The newly generated/evolved data should be accommodated in the learning process. The learning decisions should not be wholly dependent on the previously learnt aspects. The outcomes of the learning can change or get impacted with this new data. The model generated needs to be updated with this new data.

Traditional approaches generally discard the previously learnt concepts while trying to accommodate the new data. A very common factor that is accounted in these traditional learning is *catastrophic forgetting*. This states the fact that the learning methods tend to forget everything that is learnt previously while trying to accommodate the new data. If we talk about supervised methods, they would definitely suffer from this issue and spend substantial time in the re-training which is not desired. Such re-training can result in incorrect decision-making impacting the analysis. IL tries to address this issue.

Many questions arise when we say that IL needs to accommodate the new data, for example:

1. Should the entire new data be considered for learning and building the model?
2. What will happen to the knowledge base that is generated?

A very important and peculiar characteristic of IL can justify and address this, i.e., to be *selective* in nature. IL methodology needs to be selective in terms of the data selection—which data is to be used for the learning process. Essentially, knowledge amassing should occur in this process but this should be precise and selective as well.

Adapt: By having adaptive property, IL tries to adjust and behave with respect to the dynamic environment and thus being more selective in terms of the new data. The time or rather the rate at which the learning system needs to be adaptive to capture the changes in the environment is important to avail this adaptive feature.

This property is left unnoticed by the traditional approaches that can affect the decisions to a large extent.

Evolve: The learning model needs to evolve on its own. It essentially combines the adaptive and accommodative feature by being selective. It evolves with respect to the knowledge being formed. This evolution needs to address cases where:

1. Few aspects of previously learnt data are only required for current decision-making.
2. Few of the learnt scenarios or classes are not required in the newly developed model.
3. There is an overlap in the decisions, leading to merging or removal of scenarios.
4. The learning process is restricted with pre-defined classes.
5. There is a need to evolve a new class/cluster/scenario altogether with the new learning.

While IL tries to achieve the above mentioned properties, a well-known issue that needs to be handled here is *stability-plasticity* dilemma. This states that a stable classifier or the learning model will be able to preserve the previously learnt knowledge completely but cannot accommodate new data, whereas plasticity states that it will be able to adopt and learn but cannot preserve the earlier learnt knowledge.

An IL model hence needs to achieve balance on the stability-plasticity spectrum.

◊ 8.4.1 Incremental Learning or Semi-supervised Learning or Incremental Clustering?

A point of concern while studying incremental learning, is the differentiation among the methods of semi-supervised learning and incremental clustering. We can always say that incremental learning performs the learning task in semi-supervised way thereby learning from labelled as well as unlabelled data. In case of incremental clustering, without impacting the previously formed clusters, i.e., without re-clustering, the IL approach builds clusters as required or updates the existing ones. It has the capacity to decide whether a merge/dissolve/generate operation is required for the cluster management.

◊ 8.4.2 Absolute Learning vs. Selective Learning

An IL approach that is based on learning from all the available data is said to performing absolute learning. Such learning cannot justify the necessity of the new learning but simply address the fact that the approach is in position to learn from new data. It is desired that the learning be selective; selective to determine the discriminating datasets and the classes or the scenarios which can impact the predictions. Moreover, it needs to be selective in terms of the available perspectives, the context as well as the content.

Thus, the capacity of absolute learning is restricted with entire learning, whereas selective goes beyond this limitation trying to make best predictions considering the available facts and figures. Thus, it identifies the essential elements to be used in the learning process. Further, a selective approach can integrate a feedback system to have the learning process more effective.

At what point of time does the learning need to perform a change and extract the required aspects is also a factor that the selective learning needs to look at. In short, it is a learning that is 'active' all the time and is in position to discover, locate and perform the learning at particular areas. Figure 8.5 details the parameters involved in selective learning.

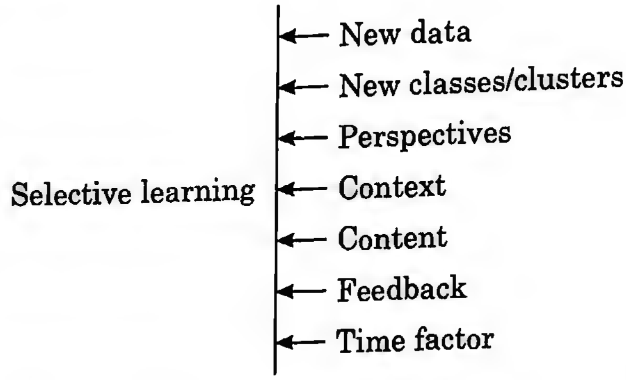


Figure 8.5 Factors involved in selective incremental learning.

◇ 8.5 INCREMENTAL LEARNING FOR KNOWLEDGE BUILDING

When we say that the incremental learning process needs to build knowledge that will be exploited in the learning, we want the learning model to be selective in terms of the modification and the updation. The learning approach here needs to make the classifier to able to adapt and mould itself and identify the drifts to perform the task of knowledge amassing. Figure 8.6 details the knowledge building aspects.

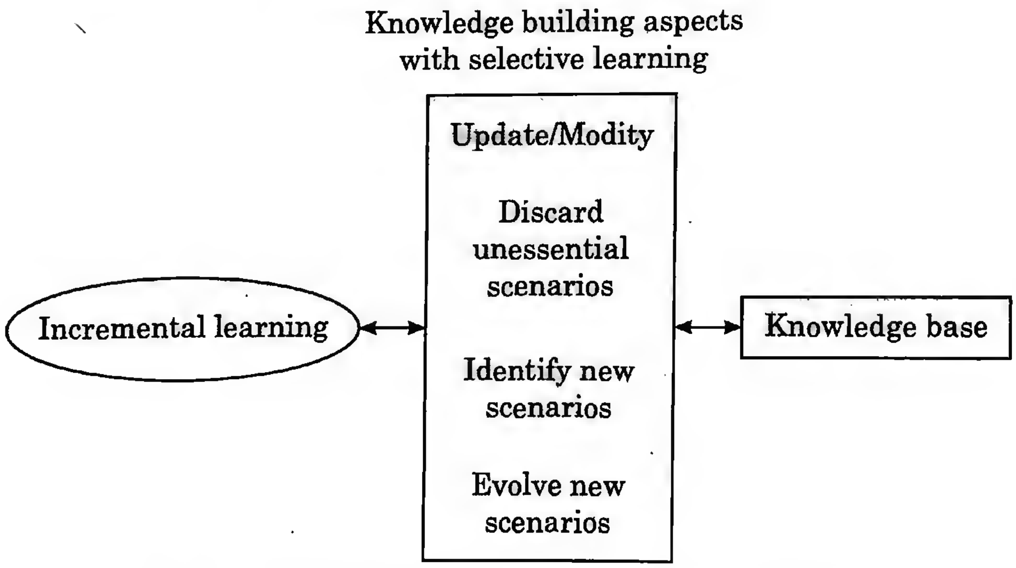


Figure 8.6 Knowledge building aspects.

◇ 8.6 INCREMENTAL TECHNIQUES TO HANDLE BIG DATA

The discussion in the Section 8.4 addressed Machine Learning techniques to be exploited for analysis of Big Data. This section highlights the need and importance of IL in the same.

The first question that needs to be addressed is why IL in Big Data? If we talk about the growth of Big Data, the rate at which this is getting generated is high. Thus, the magnitude of the records is large. The underlying data is changing with respect to time and there is a continuous flow of the data. In turn we are referring to the continuous data stream. A notion of 'Concept Drift' occurs in the processing of this data stream. Though there are many standard

algorithms that are applicable to handle them but incremental learning makes a difference as follows:

1. The working principle of incremental learning is based on the simple fact that it assumes that at initial stages very little amount of training data is being available.
2. The approach captures the relevant aspects to be learnt with the newly evolving data.
3. The utilization of the resources, memory and time is most effective, which is generally affected by the traditional Machine Learning approaches.

Figure 8.7 depicts the peculiar features that make the necessity of IL in Big Data.

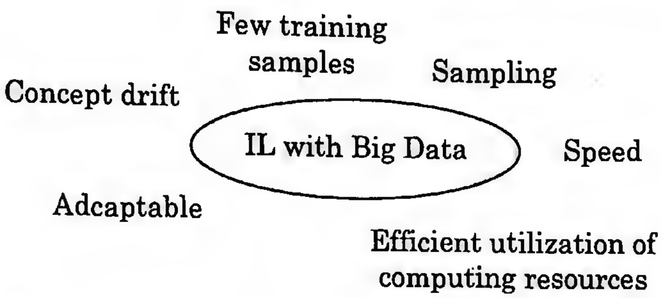


Figure 8.7 IL features for Big Data.

At present it is possible to convert the traditional approaches for incremental learning. Incremental Naïve Bayes, SVM, Decision trees and NN exist for this learning from the Big Data. Moreover, there can be Gradient approaches too exploited for the learning features.

In order to deal with the continuous data stream, ensemble-based methods are also used. A point of differentiation between them is that ensemble approaches can discard the learnt aspects, whereas the incremental one can work on selective terms. The learning paradigm, IL can be used in batch approach thereby sampling the datasets to perform the online learning.

◇ 8.6.1 Characteristic: Online Learning

The 'Incremental Learning' method needs to be active throughout. It has to adapt and update the built model. This can happen only if the learning approach is 'online'. In this section, let us understand how an online IL method would perform predictions for the incoming data stream. Figure 8.8 shows the working model for online learning.

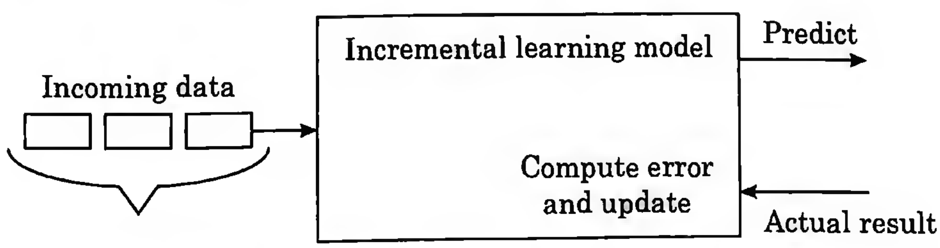


Figure 8.8 Working model for online IL.

Let us assume that datasets x_t, x_{t+1}, x_{t+2} ... is the incoming data stream.
Let y_p be predicted response and y_a be the actual output.
The error calculation would be: err(y_p, y_a) → KB

Thus, the KB (Knowledge Base) or the model is updated and the learning approach performs online—active learning. The online learning model can have variants with respect to the learning the error which could be in the form of weight adjustments to learning from first and second order information with gradient approaches.

There are many advantages of this approach, to mention a few:

1. Scalability and efficiency is high.
2. The model works on the principle of update and learnt.
3. It can be easily applied in distributed environment.
4. Can be exploited in parallel.

◇ 8.6.2 Incremental Approach and MapReduce

Till now the discussion gave us the overall idea about how IL can be applied for Big Data. Let us now consider the aspect of using incremental approaches in MapReduce. MapReduce, as we are familiar, is a model that is used for data-intensive applications. How can an incremental technique be applied here? Let us begin with some generalized aspects. More or less it is concentrated towards incremental update. There are varied approaches to carryout this task of incremental processing of the data. One such approach could be: simply treating the entire batch of data for parallel processing and learning it incrementally. But naturally this is not what is desired when we talk about Big Data. Other approach relies on incremental algorithm that we were discussing in the previous section. Here the entire dependency lies on the complexity of the algorithm that we would use and the task wholly lies on the developing and building a powerful approach that would process and compute the data efficiently for analysis while capturing the new data. One more approach that is commonly discussed in incremental processing is referred to as continuous bulk processing where the dependency lies on the application and the model needs to be changed with change in the application. The method is programmer centric and relies on the programmer to improve the efficiency of the method. This approach deals with the continuously growing data where the computational data that is impacted owing to the changes in the input are treated. The method is particularly observed and used in search engines. Few issues identified in such approaches are their inability to carryout the tasks in transparent way, requires Modelling to a new programming paradigm and thus affects the computation complexity. How can this be overcome?

For incremental processing with MapReduce, two categories exist for doing this processing. One that deals with modification of HDFS and the other that uses HDFS without modification but needs repartitioning of the state data.

One such approach is IncMR that deals with incremental processing of the data where the job submission differs stating how incrementally the newly evolved data would be accommodated and the same time, it needs to discover the new patterns to learn. The other approach Incoop suggests on making modifications to the HDFS to have the incremental processing. This is required for the incremental data discovery and for storing the results obtained in the intermediate processing.

Typically, in such systems, for incremental HDFS, the way the chunk formulations take place affects the incremental approach. Here content-based chunking is used that is able to

cope up with the incremental changes, thereby letting the inputs to the MapReduce to be stable. Further, for incremental Map and Reduce operations, memorization of the tasks avoids the running of the different splits in case of Map which are computed previously. Whereas, in case of Reduce, the approach produces key-value pairs for grouping purposes. Incremental approaches uses job scheduler and depends on the re-partitioning of the state data.

To summarize, these approaches deal only with incremental processing of the data that would be executed with MapReduce trying to address the issue of 'transparency' and 'efficiency', yet relying on memorization at some or the other levels of implementations. They can be made more potential to address the analytic technique by having the algorithm of 'Machine Learning' itself to be clubbed with this as 'incremental'.

◊ 8.7 APPLICATIONS

Big Data analysis with Machine Learning essentially is required for every application domain. From a simple recommender system to social network data or even the sentiment analysis, machine learning is need of the time. From the different learning paradigms discussed, the incremental approach is now what is looked at for improved analytics. If we consider the social media, the incremental approach can be utilized for aspects like giving recommendations. Unlike the traditional approach which would work by providing ranked suggestions, the IL will capture the response from the user as an action to the suggestions and study and evolve the same to be utilized in the next learning phase.

Let us consider a simple recommender system for book, the incremental approach would observe the trends and patterns with respect to the data/books being purchased and perform learning based on the available information and this purchasing pattern. Every new activity that takes place would be captured and the model of IL would update the relevant aspects observed from this.

So, if we try to relate it in terms of Mahout, the learning algorithms built can be extended to work incrementally to provide improved recommendations and better business models.

It is not just a recommender system but any Big Data processing which now requires the data to be processed with incremental update to treat and accommodate the new evolved data, and at the same time, it captures the results and evolves with accurate results.

◊ SUMMARY

Machine learning with incremental learning is the necessity today to deal with the Big Data owing to the requirement of accurate predictions at a faster rate and on top of it to be adaptive with respect to the environment. Further, the environment can be distributed where the existing approaches can be extended to perform the analytics with MapReduce. The incremental approach needs to perform an online learning thereby considering the trends and proposing or predicting decisions. The system has to learn continuously based on the feedback and propagate and update the same in the environment it is working.

Nevertheless, there is a huge scope of work with statistics and probabilities for the working model of IL and enhance the existing algorithms capabilities to deal Big Data efficiently and generate predictions and recommendations on the same.

Multiple Choice Questions

1. The performance of a supervised learning can be impacted by:

   (a) Training data (b) Testing data
   (c) Unlabelled data (d) All of them

2. Learning paradigm that does not depend on the availability of the entire labelled data at once and can add new data selectively is:

   (a) Deep learning (b) Multi-perspective learning
   (c) Supervised learning (d) Active learning

3. Which of the following is true about selective incremental learning approach?

   (a) Active approach all the time
   (b) Relies only on context
   (c) Accommodates entire new data as available
   (d) Has ability to change and adapt

   (i) (a), (c) (ii) (a), (d)
   (iii) (a), (b) (iv) (c), (d)

4. Which of the following is true about the stability-plasticity for a classifier?

   (a) A stable classifier can adapt new data easily
   (b) A stable classifier can preserve the learnt knowledge
   (c) Plasticity feature allows to adapt to new changes

   (i) (a), (b) (ii) (c)
   (iii) (b, (c) (iv) None of the above

Concept Review Questions

1. Describe the basic categories of machine learning.
2. Explain active learning.
3. Discuss the importance of the properties that an incremental approach needs to possess.
4. Explain the factors involved in selective incremental learning.
5. What is 'online learning'? In what way can an incremental approach be useful to this learning paradigm?

Critical Thinking Questions

1. Assume a set of dataset being collected about performance of student. A learning algorithm has to predict the performance based on the Machine Learning techniques. Which learning paradigms would one consider for the predictions? Justify.
2. Explain in what way Deep Learning's layered approach is useful. Think of any application and discuss.
3. Can there be any drawbacks of performing adaptive learning in Machine Learning? Discuss.

Laboratory Assignments

1. Consider twitter data for Nestle Maggie noodles news and identify trend of twits by assuming certain time interval. Apply any machine learning approach for the same.
2. Perform clustering on any historical data and work on incremental update for newly available data. Identify the changes in the formed clusters by the new learning.

9

Analytics in Today's Business World

—META BROWN

◇ 9.1 INTRODUCTION

Business people often trust personal intuition, more than quantitative data or other concrete evidence as a basis for decision-making. The business press is loaded with tales featuring an entrepreneur or executive who made a decision that went against all evidence, yet the outcome was good. These success are always attributed to the superiority of intuition over data, never to chance.

Failures of intuition-based decisions do not appear in the news; nobody hires a publicist to spread stories about business failures. And the business world is very tolerant about publication of unverified claims. Spinning a story in the most positive light is not merely allowed, it is expected. News about triumphs of intuition in the business world is edited, exaggerated, and sometimes simply fabricated to maximize appeal to readers.

◇ 9.1.1 Business Value of Analytics

Nearly all large businesses use analytics, though the details vary a lot from one company to another. Market research is a common application. In a 2008 interview, Fortune senior editor Betsy Morris quoted Apple, Inc. Co-founder and Chairman Steve Jobs stating, 'We do no market research'. This quote was embraced by many aspiring technology leaders, who used it to justify business practices such as developing products without evidence of market demand. Visionary leaders, they reckoned, have a better sense of what prospective customers want than the customers themselves.

Documents released in conjunction with a lawsuit later revealed what Mr. Jobs would not: Apple does market research. It collects and uses data to understand the likes and dislikes

of consumers. Analytics has a role in Apple's extraordinary success. Why would an executive deny that the success of his business depended on collection and analysis of data? It was not ignorance; he certainly knew what went on in the company that he led. Perhaps he liked to be seen as a business prophet. Perhaps he wanted to mislead competitors and kept the competitive advantage gained from analytics to himself!

Critics may doubt that Apple's success makes a satisfactory case for the business value of analytics. Indeed, it is easy to find many better examples:

- No industry leverages analytics more seriously than insurance, which developed in tandem with relevant mathematics such as probability theory. India Brand Equity Foundation estimates the value of the insurance industry in India at more than 66 billions USD for 2013.
- Search advertising, a key text analytics application, represents 30 per cent of India's digital advertising market, according to Internet and Mobile Association of India. This puts the value of search advertising at around 11,000 crore rupees for 2015.
- The telecommunications industry is highly competitive, and providers around the world depend on analytics for clues to acquiring, retaining and increasing revenue from customers. Grameenphone, Bangladesh's leader in telecommunications, reported that a pilot customer churn prediction project led to a campaign take-up rate of over 20 per cent (compared to 3 to 5 per cent with earlier campaigns) and an increase in customer revenue as well.

Analytics provides the best source of guidance for good business decisions. Thoughtful use of analytics has enabled diverse businesses around the world to yield fortunes in revenue and profit. Even those successful business leaders, who publicly boast of their personal intuition for decision-making, are quietly using analytics behind the scenes.

◇ 9.1.2 Limits of Intuition

As a reader of this book, you are probably already convinced that analytics has value. Yet you may not find it easy to make a persuasive case for analytics, one that could win over your manager, a prospective client, or a friend. People resist analytics for many reasons:

- **Confidence:** Belief in their own understanding of the consumer.
- **Hindsight:** Rationalizing past prediction failures.
- **Fear:** Concern that power or creative freedom will be lost.

Direct response marketers (those who sell directly to the consumer) have been making good use of analytics for about a century. David Ogilvy, cofounder of the advertising firm Ogilvy and Mather, and later Chairman of Ogilvy and Mather India, was perhaps the most influential advertising expert of the late 20th century. He spoke of two worlds, direct response advertising and general advertising, and explained that direct response advertisers have a great advantage over general advertisers, because they know what exactly what kind of advertising works (see video of David Ogilvy himself explaining this at https://www.youtube.com/watch?v=Br2KSsaTzUc). They do not guess; they know! They know because they test alternative ads and measure the results.

During the 2012 United States presidential election campaign, the Obama for America campaign team tested alternative versions of all its email solicitations for contributions. One of the most important things to test in email advertising is the subject. If the subject line is not interesting, the email would not be opened. For example, tests revealed the subject 'Hey' was remarkably effective, so that was used often in subsequent email ads. An integrated program of analytics for direct marketing combined with micro targeting (market research and campaign methods focused on tailoring messages to individuals) enabled the campaign to raise more than 1 billion USD.

If that story is not enough to persuade your manager that analytics outperforms intuition, use a demonstration. Test some alternative ads for your own work, and invite the manager to predict the results (If you do not have ads of your own, you can find many examples, complete with test results, at Which Test Won, https://whichtestwon.com/). Present a set of examples, perhaps ten or twelve pairs of alternative ads, and ask the manager to pick the version of each ad that will work best. Record the answers and compare the predictions to actual test results. This exercise has shown many confident business people that their intuition was no better predictor of consumer behaviour than the flip of a coin!

◇ 9.1.3 Aligning Analysis and Action

Merely analyzing data produces no benefits. The costs and effort invested in data collection and analysis lead to returns only when the resulting information is put into action.

Connecting data analysis with action presents certain challenges for data analysts. As a data analyst, you must identify important business problems, develop an understanding of the range of possible corrective actions, and plan appropriate analyses to determine what action is most appropriate. You must prepare, present and defend the business case for analysis. And you must present results persuasively.

◇ 9.2 BUILDING THE BUSINESS CASE FOR ANALYTICS

IT investments often fail to produce good returns. A 2012 report by Gartner, Inc., a technology research firm, found that 20 per cent of small IT projects (defined as those with budgets under 350,000 USD) failed, and that the bigger the budget, the more likely the failure. The executive who controls the funding that you need may well view analytics as just another IT project. You would not get the money just by asking. It will be up to you to prepare a convincing case for the investment.

Every business case has two major parts, costs and benefits. Outlining costs is straightforward, since you know what products and service you want and what they cost. You may also need to account for internal costs, such as staff compensation and overhead. Defining benefits is not nearly so simple.

Benefits take one of the two forms: revenue increases or cost decreases. Revenue benefits may seem the most tempting, since the potential to increase revenue is, in theory, unlimited. The problem is that managers with budget authority are often wary of promised revenue benefits. They are more comfortable with projects aimed at reducing costs. This is not just doubt about

the outcome of data analysis. To achieve a revenue increase, the business may require several managers to take action, so the person who controls the IT budget may not have authority to make the business changes required to realize revenue increases.

◊ 9.3 DATA ANALYST'S COMMUNICATION CHALLENGE

Data analysts face a special challenge while discussing their work with decision makers. Our training and much of our work, focus on model accuracy and precision, choosing appropriate tests, and other technical concepts. These things are of no interest to business executives. It is not up to executives and clients to learn the data analyst's language! Data analysts must explain analytics in the language of business. When presenting to executives, keep the following guidelines in mind:

- **Use financial language:** Do not speak of test statistics or statistical significance. Instead, describe your results in terms of money (such as number of rupees, dollars or euros) or closely related terms (such as per cent increase in sales, conversion rates, or customer churn rates) that are familiar to the executive and obviously connected to the financial well-being of the business.
- **Be brief:** Executives are busy! Present the most important information first, and keep it short. Make the first minute interesting, and you will be rewarded with a little more time and patience from the decision-maker.
- **Minimize detail:** Focus on important points and omit minutiae. Most executives find these dull and irrelevant. You don't want to risk losing the decision maker's attention, and there is an even worse risk, that your listener will become so intrigued by some small and unimportant element of the presentation that your main points will be neglected.
- **Reveal specifics gradually:** Do not share everything you know at once. Plan to begin with a few major points and then present supporting information a little at a time. Leave some obvious openings for questions, and be prepared to change the order of your presentation, drop some topics and emphasize others in response to the interests of the decision maker.

Make your presentations easier to understand by telling stories that reveal what you have discovered through data analysis. Do not feel that everything you have to say must be expressed in story form! Instead, use one or more brief stories within your presentation and relate the rest of the presentation back to the story.

A marketer at a digital advertising agency placed ads for her client's product on a social media site. The ad did not result in many sales. She checked her web analytics reports and found that many people were clicking the ad, but the bounce rate was very high. In other words, people who clicked on the ad left without buying the product. More careful review of the report revealed that this happened only on the mobile site. Finally, the marketer did some testing and discovered that the mobile site was not working properly. The customers were not buying because it was impossible to complete a purchase on the mobile site.

◇ 9.4 STORY-TELLING WITH DATA

Data analysts tend to relate such discoveries saying things like, 'I reviewed the report', 'I ran a test,' and 'I found that....'. In other words, data analysts often refer to themselves and their own work. But this marketer knows that the client is not interested in her, or her work. Clients do not want to know about you, they want to know about their customers! So, the marketer tells a story like:

> Anil was reading updates from his friends when he noticed a sponsored post showing an image and the price of your new game. He was delighted, and ready to buy the game immediately, so he clicked on the ad. But when he tried to fill out the payment information on your mobile site, he could not enter any information. He tried reloading the page several times, but he still could not enter the information. Anil gave up, and never bought your game.

By opening a presentation with a story like this, you can engage listeners and make them more open to listening as you present supporting data. The whole story is only a few short sentences, but it clearly explains the business problem, it is easy to understand and remember, and it is interesting, since the client is deeply motivated to sell the product. With very few words, you can include all the basic elements of a story:

- **A protagonist (hero):** The story must be about someone (usually a customer), and that someone is not you (the data analyst)!
- **A challenge:** The protagonist has a goal, and faces an obstacle that must be overcome to reach that goal (In the movies, heroes face a series of obstacles, but your stories should have just one).
- **An ending:** Does the protagonist achieve the goal, yes or no? (Anil did not! But you will use data to show what action the client must take to change the situation so that the story can have a happy ending next time.)

And data stories have one special requirement: they must be true. Your data must show that the sequence of events described in the story actually happen. (Your story will be most compelling if it can be told by a real customer. Perhaps you have audio of the customer's voice from a technical support or customer service call, or a message from the customer that you could read aloud in the presentation. If you have the opportunity, you might record short video interviews with customers to use in presentations.)

◇ 9.5 TEAMING WITH COMPLEMENTARY ROLES

No data analyst has the skills, or the authority, to do everything it takes to make an organization data-driven. It is a team effort, so you must get familiar with complementary roles and the people who do them. Data analysts must have good working relationship with following professionals:

- **Executive management:** The impact of your analytic work depends on your ability to understand the concerns of executive management, and make a convincing case for action based on the work you have done.

- **Information Technology (IT):** Access to data and business systems is controlled by the IT team. Many data analysts resist working with IT, but that is foolish. You will need a constructive working relationship with IT to get the data that you need and the resources required to integrate your findings into business applications. Respecting IT standards also prevents you from violating privacy laws and other legal obligations.
- **Business analysis:** These are your organization's change management experts. They help your organization to improve processes, and avoid costly and unpleasant mistakes along the way.
- **Project management:** Project managers lead planning and execution to complete a defined task, such as implementation of a new business system, or constructing a building.
- **Subject matter experts:** You cannot do much with data if you do not know what it represents, how the business works, or what might be done with your results. If you do not have relevant knowledge of the business yourself, you go to subject matter experts for more information. This is not a job title; it is a role that might be filled by anyone who has knowledge that you need.

◊ 9.6 LIMITS OF ANALYSIS

It is easy to become so confident of a particular analysis method, data source or result that you lose sight of the limitations of your own work. Overconfidence adds to the risk that you will draw an unrealistic conclusion from your data, mislead a client, and end up with a bad outcome for the business. You have a responsibility to examine your own work closely for errors and limitations, document your processes in detail, and invite peer review. Consider following areas:

- **Data:** Is relevant data available? How have you assessed the quality of the data? Are you certain that you know how the data was obtained and what each field represents? What are the limitations of this data (for example, data collected online may not be representative of individuals who do not use the Internet)?
- **Analytic methods:** Are the analysis techniques that you intend to use appropriate for the data and the application that you have in mind? Do you have adequate tools to conduct the analysis? Has the data been properly prepared?
- **Trust:** Have you complied with applicable data privacy laws and other legal obligations? Does your intended use meet ethical standards (and do you know what standards your employer, licensing authorities and professional association prefer?) Who owns the data, and will the owner be comfortable with this use of the data?
- **The analyst:** Do you know the underlying assumptions of the techniques that you are using, and have you verified that your assumptions are reasonable? Can you explain your reasons for selecting specific data sources and analysis techniques? Have you followed an accepted analysis process? Can you relate your results to action?

◇ 9.7 IDEALISM AND REALISM IN BUSINESS ANALYTICS

Many parents encourage their children to become physicians because medicine is a respected profession that ensures steady employment and a comfortable income. These reasons are legitimate, yet a career in medicine also involves dealing with down-to-earth things such as blood, mucus and urine. In recent years, a career in data analysis has acquired certain glamour. You will certainly find great opportunities in the field, yet you will not always find it simple to get the resources you need or achieve the impact that you may wish.

◇ 9.7.1 Interplay of Culture and Analytics

In statistics classes, we are taught to evaluate the value of an analysis based on accuracy, precision and other technical grounds. Yet technical sophistication and excellence are of little importance to satisfy the wishes of business executives. The data analyst who will have the most impact in the business world is not the one who focuses most on accuracy, but the one who best matches the analytic process to the preferences of powerful decision-makers.

There is no single best approach to performing analysis and presenting results; you must tailor your work to your own environment. Management styles vary by country (In the individualist culture of the United States, executives usually make decisions as individuals, while Japanese are far more interested in consensus), industry (Bankers are slow to change processes, while certain technology businesses are far less resistant to change) and the individual manager.

◇ 9.7.2 Why Business is not Data-Driven

You may have read business best-sellers or news reports which showcase success stories of companies who have profited by using analytics. Perhaps you recall a story which was very impressive, one that you would like to emulate. Now, you should investigate another side of that success story.

◇ 9.8 READING BETWEEN THE LINES OF SUCCESS STORIES

Find someone who works with a company, and start a casual conversation. It does not matter what job or department that person holds, just listen. The conversation will come around to work, and you are bound to hear some details that you would not find in any book or news report. A business that has analytics competency and made excellent use of analytics in some functions may still fail to use analytics to address problems in other areas. You might discover that waiting times for customer service are far too long, that a new product has a quality problem, or that staff turnover is awfully high. Remember, you are investigating a company known for its outstanding use of analytics; imagine how much worse things must be elsewhere.

◊ 9.9 RELUCTANCE TO USE ANALYTICS

Most data analysis methods used in the business world are not new or secret. Some can be implemented with nothing more than paper and pencil, and many can be put to work with the power of an ordinary desktop computer. The details are available at most public libraries and on the internet. Every executive on Earth has access to this information, so why do not they all use it? This is due to the following issues:

- **Analytics is challenging:** It may be no more difficult to perform basic statistical analysis than to do good accounting, but there are legal and contractual obligations that force business to use proper accounting practices. Data analysis is rarely required by law.
- **Analytics requires data:** There may be many obstacles to data collection and access.
- **Analytics implies transparency:** Data-driven decision-making implies admitting that some things are not working.
- **Analytics must be tied to action to be valuable:** Many managers do not feel comfortable committing to choose action based on analysis.

◊ 9.10 BUILDING TRUST IN ANALYTICS

Help executives to feel more comfortable with data-driven decision making by introducing analytics gradually. Start with small projects that do not require large resource commitments. Success with these projects builds trust that you will need to secure funding for more elaborate works in the future. Choose low-risk projects at first (for example, comparing a free shipping offer to a modest discount). And, at every stage of the work, speak the executive's language. Explain goals and findings in monetary terms aligned with the executive's responsibilities.

◊ 9.11 IMPACT OF BIG DATA

In 2001, Doug Laney of META Group, a research firm (later acquired by Gartner), outlined the challenges faced by some of his clients in dealing with modern data sources. He summarized the issues in just three words:

- **Volume:** The quantity of data collected is extremely large.
- **Velocity:** Data is collected rapidly.
- **Variety:** The data is in diverse formats.

Laney's article was the seminal description of what we now call 'Big Data' and his succinct description of Big Data challenges has been embraced by the analytics and business communities as the '3 Vs' of Big Data. His words have been so influential that they are now far better known than Mr. Laney himself. Despite advances in computing technology, the same issues continue to challenge businesses today.

◇ 9.11.1 Is Big Data New?

If you view the challenge of Big Data as unique to modern times, you may wish to ponder a bit of data history. The first United States census, conducted in 1790, called for dispatching paid data collectors across each of the states, to locate and record basic information about each and every person in the country, over the course of nine months. That was just to collect the data, handwritten on paper. These data collectors did not even have standard paper forms to use. Imagine the effort required to tabulate the results! Every generation confronts its own data challenges.

◇ 9.11.2 Big Data: Where Does It Come From?

Three primary sources create the bulk of Big Data. These are as follows:

- **Conventional business activity:** Records of transactions and other everyday business and government activity, as well as research data, fall in this category. This is the information that was collected before the Big Data era, and even before the computer, yet now it is gathered in larger quantity and in great detail than ever before.
- **Computing activity records and user-generated content:** Social media posts, email, SMS, web activity logs and other data generated in the course of online communication.
- **Machine monitoring:** Information recorded by sensors in machinery in industry and public settings, including surveillance records, data from the 'Internet of Things', and airplane flight recorders.

The diversity of data within each of these sources means that one organization's Big Data challenges may be a world apart from another's. One data analyst may be challenged with searching for the face of a known criminal amid a million hours of surveillance video, while another seeks the clues to find prospective customers in short text posts, and still another reviews routine business records in search of fraudulent transactions.

◇ 9.11.3 Pressure to Derive Value from Big Data

At a recent analytics conference, a speaker representing a Big Data software firm displayed a very grainy photograph. The image was barely recognizable as a face, but it was not possible to say for certain whether the face was that of someone male or female, young or old, let alone recognize any individual person. The speaker replaced the image with more and more finely-grained versions until at last it became a clear and detailed image of one specific woman whose appearance and demeanour could be clearly seen in the photograph. This, the speaker said, was the effect of Big Data. While that vision is appealing, not many organizations have yet achieved success with Big Data analytics, and not every Big Data source offers has quality, richly detailed data.

It is certainly true that there is value in detailed and accurate data, but you should not assume that every Big Data source offers accuracy or valuable detail. The best data sources

provide data that is accurate and relevant to the problem you seek to solve. The mere size of a data source is of no value whatsoever, from an analytical point of view. Massive quantity implies great data management complexity and cost, but not necessarily great business worth. It is the data analyst's responsibility to thoughtfully evaluate the suitability of any data source for a particular application.

Those who possess Big Data are under tremendous pressure to do more than merely maintain it. Collecting and storing that data are expensive; your management wants something in return. You may not be able to produce something valuable from every data source you encounter, but you can easily learn the characteristics that indicate the best potential for producing valuable results.

◇ 9.11.4 Making Big Data Pay

If you could observe each prospective customer personally, you would learn many facts that would help you to sell to that person. If you watched a customer say Priyanka Singh as she shopped, you may see that she bought items for baby care, food, and cleaning supplies. Your observations would give you clues that Priyanka is a value-focussed shopper who is caring either for her own family, or someone else's. You might start a little chat with her, and learn that she is a homemaker, that she has a baby daughter and a three year old son, and that she also cares for a neighbour's baby son. Knowing all of these, when Priyanka comes to shop next time, you might direct her to some fresh in-season produce that is on sale, toys suitable for preschool children, or a new type of diaper. You would not waste your time or her, telling her about office supplies, because you know she has no particular reason to be interested.

◇ 9.11.5 How to Identify Valuable Big Data Sources and Opportunities

The key to make Big Data pay lies in using data to simulate what you would do if you personally observed each individual customer (Similar opportunities are also found in government and non-profit applications, although the rewards are not necessarily money). So, a typical opportunity to profit through Big Data would be found in online direct marketing. Online retailers do online direct marketing, as do non-profit and political fundraisers. These applications are common, so opportunities are abundant. To profit from them, you need the right kind of data. What matters is not so much the size as the suitability and richness of the data source. Desirable Big Data sources have the following characteristics:

- **Relevance to a specific business problem:** If the object is to sell to a person, the data source must include the same kind of information that you would learn by watching the person in a conventional shop. Who is this person? When does she come to the shop? Has she made a purchase? What did she buy? Is she a repeat customer? Does she pay full price or look for discounts?
- **Detail:** You must have information about individual people and individual transitions. Aggregate data will not do.

- **Quality:** While it is futile to look for perfect data, information that is largely incorrect is useless. Make careful use of data quality checks to evaluate each dataset before you invest time on analysis.
- **Availability:** Updated data must be available on an ongoing basis for predictive modelling use.
- **Path to action:** The data must include some identifiers that allow you to take action. You do not necessarily need the customer's name, but you must have a way to get the offer to the right person.

◇ 9.11.6 Big Data Working Environment

Remember, the larger the budget of an IT project, the greater the risk of failure. Manage risk in Big Data analytics by resisting the urge to start on a grand scale. Start with small-scale, low-risk projects. How should you do that when the scale of the data is massive? There is no need for exotic analytic methods. Just limit the project scope (for example, look at one product instead of all the products you sell), and use modest samples of data in the beginning.

You can carryout starter Big Data projects using most of the same processes and methods that you use for any other type of data analysis. The most significant difference between a typical data analysis project and your starter Big Data project will be that obtaining a proper data sample may require a more complex process (Your counterpart in IT is not an expert in statistical sampling techniques! Plan to provide detailed instructions). Also, give thought to the scalability of the analysis methods you choose. Some analysis methods are impractically slow when used with large amounts of data. Remember that you can often develop models with small samples of data, and only use the larger datasets for scoring, which require far less computational power.

◇ 9.11.7 Big Data Demands Constructive Teamwork

When working with small amount of data, some data analysts cheat. They do not go through the proper channels to obtain data. They do not document their work well, and sometimes they do not document at all. They store data, computer code and reports in odd places and do not share with everyone who should have access. They use the wrong tools. They do all these things and more, all bad business practices. And often, they get away with it. The work may not be as good as it should be, and it may not have the impact that meaningful data analysis should have, but most of these cheating data analysts do not lose their jobs or suffer complaints from their managers.

When you work with Big Data, you cannot cheat and get away with it. You cannot hide a Big Data source on your laptop. You cannot obtain the data at all without going through the proper process. Your work will draw attention, and you will be expected to do more explaining and produce better documentation. Using the results means that your models must be integrated into business systems, something you cannot do alone. Big Data success depends on teams of people with diverse skills and responsibilities working together to achieve shared goals.

◇ 9.12 RISING IMPORTANCE OF TEXT IN ANALYTICS

As recently as the 1990s, text analytics would not typically have been mentioned at a gathering of data analysts. Text analysis was not part of a statistician's training. Social scientists did use text, but the process was manual, and businesses often delegated it to outside specialists. In the later part of that decade, text analysis tools were becoming available, but drew far less attention in the analytics community than the rising field of data mining. Still, one text analytics application, online search, was coming into widespread use, and becoming a part of everyday life.

Today, there are hundreds of text analytics products, conferences for both academics and business users, and every up-to-date data analyst is aware of text analytics. The public has heard about it from the *Times of India*, *The Economist*, and *The New York Times*, and other media sources.

◇ 9.12.1 Unstructured Data Resources

A growing share of the information stored in electronic data formats is unstructured, that is, not in traditional data formats such as numbers and categories. As the cost of electronic data storage plummets, more of it is devoted to following forms:

- **Video:** Security and surveillance, research, communication, entertainment, personal use
- **Audio:** Call monitoring, communication
- **Image:** Identification, personal and business photography, reconnaissance, medical imaging
- **Text:** Messaging, documentation, chat, news, email, help requests, warranty claims

Of these, text draws the most attention from data analysts. Many text sources are understood to hold useful information. And although text analysis may be more challenging and imperfect than conventional data analysis, it is still simpler in many respects to work with text than other forms of unstructured data.

◇ 9.12.2 Awareness of Text Analytics

It has been only a short time that powerful computers have been available to ordinary workers. Computers of 1960s were extremely expensive and had only a few hundred kilobytes of memory, less than enough to store just one of today's photographs. By 1990s, many office workers had access to personal computers, yet these still had barely enough capacity to manage word processing and storage for everyday business documents of that era, and those documents were compact by today's standards. It is only been a short time that computing has been cheap and accessible enough to make computerized text analysis possible.

Now that text analysis is feasible, we have faced several major arguments for using it:

- **Relevance:** Information trapped within text sources is useful for addressing a wide variety of business problems.
- **Obligation:** Large investments have been made to create and maintain text sources and investors demand returns.
- **Awareness:** Text analytics technology is becoming less expensive, and more effective, accessible and visible.

Yet we still have a long way to go before text analytics becomes a part of the average data analyst's work.

◇ 9.12.3 Challenge of Demonstrating Value

In a 2014 text analytics market study by Alta Plana Corporation, 42 per cent of text analytics users reported that they had achieved positive return on investment. Stated another way, more than half of text analytics users are making no money for their efforts!

A casual review of the promotional literature for text analytics products hints at the cause of the problem. Benefits mentioned are rather vague: insights, trends, knowing the customer. What exactly is the value of these things? What actions might an executive take to convert these things into concrete, measurable returns? Simply put, buying software and hoping for insight is not a satisfactory plan.

Treat text analytics like any other business investment. Ensure positive returns by starting with a proper business plan, a document that describes a problem and its impact on the business (the costs associated with the problem), a proposed solution, and the costs and benefits associated with that solution. Benefits must be expressed in financial terms, and they may, in theory, be either revenue increases or cost savings that offset the cost of the solution. In practice, though, solutions that offer benefits of cost savings are more appealing to many decision makers than promised revenue increases.

◇ 9.12.4 Text Analytics Applications that Pay

The best candidates for text analytics applications that yield positive return on investment are those which reduce known costs for work that is unavoidable. It is easier to recognize the potential applications when you are already familiar with some of them and understand their characteristics. Here are ten good examples:

- **Coding:** Categorizing open-ended survey responses. Businesses which use these responses often send the data to outside firms for coding, a process which is slow, costly and often yields inconsistent or poor-quality results.
- **Translation:** Manual translation requires skilled human language experts. Time and cost pressure often make the use of qualified experts infeasible. Automated translation, though imperfect, produces quick and consistent results.
- **Technical support:** Live technical support calls are costly and usually require the customer wait for service. Applications that enable the customer to find satisfactory information automatically reduce costs and waiting time.

- **Customer support:** Resolving non-technical issues automatically is also a money and time-saver.
- **Routing:** Messages entering a single communications portal often relate to a diverse selection of subjects. For example, messages to a bank might include requests to open new accounts, loan applications, journalists' inquiries, service complaints and many other topics, each of which is best handled by a different department of the bank. A large bank might would need several full-time staff just to sort the messages and route them to the proper departments for handling. Automated routing reduces costs and eliminates routing delay.
- **Content monitoring:** Chat and other social media applications require monitoring to ensure that inappropriate content is eliminated. For example, adults should not be prowling in children's chat applications, yet it is difficult for human monitors to keep up with tremendous numbers of online conversations taking place in diverse languages. Aided by text analytics, monitors could work with far greater speed and completeness.
- **Churn:** Loss of customers is damaging to businesses, and the cost of acquiring new customers is significant. Any application that helps to identify at-risk customers early, while action may still be taken to save them, has business value. (Han-Sheong Lai of Paypal, a financial services company, has presented his successful work using text analytics to identify at-risk customers. Lai took a simple and effective approach to find these customers, by searching for messages with direct statements of intent such as, 'I will close my account.').
- **Lost sales:** A more subtle phenomenon than customer churn is the loss of potential sales from an active customer. When a customer wants to buy, but does not, the business loses money. Applications that help to identify these situations and enable the business to take action and overcome sales barriers preserve revenue.
- **Warranty claims:** Customers who return faulty products, provide valuable information about quality problems, and nearly all of that information is in text. Text analytics can make it possible to identify causes more quickly and take corrective action earlier, reducing losses, protecting the reputation of the business, and possibly even saving lives.
- **Liability and litigation:** A single lawsuit may require legal review of millions of individual documents. Applications specifically for this new space, known as 'e-discovery' make attorneys more effective and productive, and reduce the length of time required to prepare for litigation.

So, when evaluating a potential use for text analytics, ask the following basic questions:

- Is this work absolutely necessary?
- Is the cost of doing this work burdensome?
- Can the cost be significantly reduced by using text analytics?

Any application for which the answer to each of these three questions is 'Yes' is a prime candidate for positive returns on investment through text analytics.

◇ SUMMARY

Everything is driven by Analytics today. Right from decisions at small stores to decisions about procuring servers at big companies some type of analytics is utilized. With advent of advanced techniques for analytics and capability to mine big data now it is the time to leverage analytics in better way. Data tells the story and reveal hidden facts. With so much of information at our disposal we can make this data to speak more confidently. Business Analytics demands deriving value from big data to create business value. Business decision-making has been one of the major area of interest and business analytics and Big Data can help in great way. Business Analytics comes with challenges and with effective text mining and data correlation we can overcome these challenges.

Multiple Choice Questions

1. Major Big Data sources include:
   (a) Conventional business activity (b) User-generated content
   (c) Machine monitoring (d) All the above
2. Most persuasive business cases for analytics are those that primarily emphasize:
   (a) Valuable insights (b) Revenue increases
   (c) Cost reductions (d) All the above
3. An issue which has greater impact on Big Data analysis than conventional data analysis applications is:
   (a) Scedasticity (b) Scalability
   (c) Structure (d) None of the above
4. Data sources such as security and surveillance video, medical images and text messages are said to be:
   (a) Unregulated (b) Deregulated
   (c) Unstructured (d) None of the above
5. Doug Laney identified the key elements of Big Data as:
   (a) Volume, Velocity, Variety (b) Volume, Velocity, Veracity
   (c) Location, Location, Location (d) None of the above

Concept Review Questions

1. Data analysis must be used in combination with .... in order to yield concrete benefits for an organization. (Fill in the blank)
2. Why is teamwork a necessity for working with Big Data?
3. What are the two major types of benefits that an analytics project may provide?
4. The larger the budget of an information technology (IT) project, the .... the risk of failure. (Fill in the blank)

Critical Thinking Questions

1. Why might a manager resist the use of data and analysis?
2. Why does the business press tell more analytics success stories than analytics failure stories?
3. A news report makes some interesting claims about results of a business analytics project. What resources can you use to investigate those claims?

Laboratory Assignments

1. Ad results prediction

   (a) Prepare a set of 10–12 pairs of alternative ads. (Use real examples from your own workplace, if possible. If not, use samples from http://www.whichtestwon.com or another source of sample results from advertising tests.)

   (b) Invite individuals who are unfamiliar with the ads or testing methods to review each pair of ads and guess which ad will work the best. Record responses for each individual and each ad.

   (c) Compare predictions to actual test results.

2. Data story-telling

   Choose a sample data analysis project from your own work.

   Imagine the data from the point of view of the person whose activity it reflects. For example, a bit of mobile advertising data reflects the experience of a mobile user. Use all information sources available to you to describe that person. Registration data may give you details such as a name, and past user history. Other sources may tell you about the interests or demographic details of a typical user.

   Tell a story, from that person's point of view, which explains the experience reflected in the data. What was the person trying to do? What happened first? Then what? What was the result?

3. Proposing an analysis project

   Identify a real business problem which might be addressed using a data analysis technique that's familiar to you. Do not choose a grand, far-reaching business problem! Select a single, narrow, well-defined problem.

   Write a proposal of 1 to 2 pages, with the following elements, which proposes a data analysis project to address your business problem.

   - Explanation of the problem and its impact on the business (limit this to one paragraph)
   - Summary of proposed work and desired results (one paragraph)
   - Data to be used (is this existing data or will new data be collected?)
   - Team (who will do what?)
   - Project steps (include a timeline)

10

Conclusion

—Dr. Parag Kulkarni

Big Data has become a recent trend in technology. Whether it is network, whether data mining or even data management we begin to talk in terms of Big Data. Different books are written to handle different aspects of Big Data and data mining. Big Data is typically a huge size data dealing with different aspects—it may be data generated through social networking sites, or about a big event, worldwide interactions. Since it is huge, coming from all sources and dealing with larger landscape, it is assorted mix where major portion is unstructured and semi-structured. Mining and associating these unstructured Big Data deal with different aspects of data mining. This book has focussed on different aspects and challenges with reference to Big Data and unstructured data mining. Machine Learning for Big Data is different than traditional machine learning. While traditional machine learning techniques are more pattern-driven and focus on smaller part of datasets, Big Data related Machine Learning needs to be holistic. It needs to consider context. It has to deal with challenges of representation of data. Further, it should be incremental. This book has initially covered all aspects of Big Data and some of the techniques developed in due course. Context is the key for learning. Identifying holistic relationships and determination of topics and relationships among topics will help to organize and mine Big Data. The other aspects like clustering, incremental learning, multi-label association and knowledge representation are dealt in detail. Right from what value Big Data is adding to whole system and how it is bringing this value to table are the questions repeatedly taken for discussion. Having more data at disposal is a challenge—can it be harmful for delivering results? Some researchers say that they are not interested in too much of Big Data. Data has a purpose and is there for delivering value. The objective is not to process Big Data and make it available for decision-makeing. Rather the objective is to find out how to use this data to make this world a better place. Big Data is showing some potential and hence new learning methods need to be devised to make best use of it. Is it big analytics, big mining, big learning and big intelligence? Based on simple experimentation related to unstructured data let us try to apply Big Data to solve problems. Just getting excited by big term is detrimental to value creation. Building the holistic perspective to take best out of data to deliver value is the purpose of the

book. Big Data age came with potential to revolutionize all data-mining concepts. Big Data is opening new avenues for decision-making. It is a change in paradigm. From data hatred to data friendliness and data interpretations is the journey to Big Data. Is data delivering what we want from it? Magical powers of data need to be unleashed and hence Big Data paradigm needs to reach to altogether different level. The behaviour analysis and anomaly detection are promised with better results with Big Data. Hence, let us think Big Data while keeping our holistic perspective, and focus on making this world a better place to live in.

What next? Well, this is the most difficult question that still has no right or wrong answer. Big Data is increasing data connectivity and association. It is a journey to systemic perspective. Big Data with holistic perspective will bring us to systemic data. Learning efficiently and effectively, learning systemically and putting data in right perspective are the challenges. Business analytics for Big Data is going to be different. Actually, Big Data does not solve problems. We need strong learning and analysis mechanism to solve problem. Data abundance is harmful or is not still the question we are trying to answer. Big Data wave may prove to be very helpful with support from strong mining techniques to holistic learning techniques. When there is a change in paradigm, many conventional concepts fall apart and there is a serious need of research and technology to take this paradigm ahead. It is very much true for Big Data. Hence, Big Data is opening up new research and analytics areas. This continuous research will definitely strengthen the analytics and mining space to make this world a better place to live in.

Annexure I

Introduction to Hadoop
A Big Data Perspective

—Dr. Sarang Joshi

◊ INTRODUCTION

With introduction to advances in computing power with scalable, multi-core, multi-tasking and distributed architectures, it is very essential to build the software to utilize such advances efficiently. The data capturing sensor's portability and easy connectivity have brought the advances in the data storage, interpretation and the business perspective of analytic which may be called a Big Data. Apache's Hadoop is a software framework that enables distributed processing of large datasets across the clusters of computing machines using simple programming functions. The Hadoop scales its operations from one server to several multiple computing machines which offer local computing and storage. http://hadoop.apache.org is very important digital web reference commonly used for detail study of Hadoop. Hadoop 2.7.X is the latest version for the year 2014.

◊ BUILDING-BLOCKS OF HADOOP FRAMEWORK

Hadoop's basic building blocks include:

1. Hadoop Common
2. HDFS (Hadoop Distributed File System)
3. Hadoop Yarn
4. Hadoop MapReduce

Hadoop framework is described in Figure A.1.

Hadoop Common

It has all common features necessary to support other modules in the Hadoop framework. Recent release has added support to the Windows Azure Storage-Blob as a file system in Hadoop.

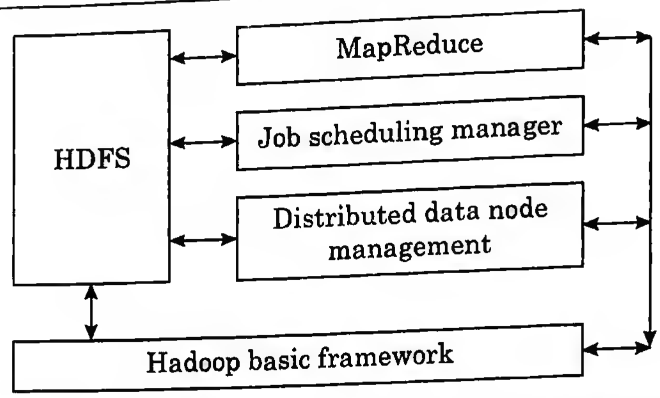


Figure A1.1 Hadoop functional framework.

These advance features operate using JDK7 and upwards. Hadoop Common is considered as the kernel or main backbone of the Hadoop Basic Framework. It is also called Hadoop Core. It initializes and activates other modules such as HDFS, Yarm, MapRedure and other installations related to the Hadoop framework with the help of JAR (Java Archive Resource) and script files. It creates necessary data structures and other system spaces necessary for establishing the communication with the computer operating system and its file system. It provides links to help other documentation and provides links to the other applications developed for Hadoop framework by the Hadoop Community.

Mini cluster is one of the important services available in the Hadoop Common. It is used to start and stop the single machine instance of Hadoop installation. This is done without the need of setting the variables or managing the configuration files. The CLI MiniCluster is started using the command on Linux or equivalent derivative platform:

```
$ bin/hadoop jar hadoop-test-*.jar minicluster -jtport <JT_PORT> -nnport <NN_PORT> w
```

In the above example command, <JT_PORT> and <NN_PORT> should be replaced by the port numbers available to the user, otherwise random free ports are to be used. The number of command line arguments can be given for this command.

Another component of Hadoop, are Native Libraries of the Hadoop Framework. These files have file extension of '.so'. For example, libhadoop.so. Depending upon the environment means, the operating system and the hardware beneath, some of the libraries can vary the installation.

HDFS

HDFS is Apache's Hadoop Distributed File System written using Java. It is designed to run on a commodity hardware that supports GNU/Linux Operating system or its derivative like fedora, Java and supports very large datasets and can easily be used with heterogeneous platforms. It is more designed for batch processing of the datasets. It is fault tolerant and it gives high throughput access to the applications data. Applications using HDFS use streaming access to the datasets rather than general purpose file systems which focus more on latency time improvements or low latency time. The datasets size under HDFS is very large, of the order of gigabytes. HDFS has features to provide high aggregate data bandwidth and can scale lots of nodes in a cluster while supporting very large number of files in a given instance. Since heavy sharing of datasets exists at the central servers, the HDFS applications preferably

use write-once-read-many access model for files. These types of files once written are rarely changed. This helps in reducing the data coherency issues resulting in fast data access. The Hadoop project applications like MapReduce and Web crawlers find this a suitable design for improving the throughput. Being distributed and suitable for heterogeneous platforms, the large dataset transfer may cause a concern hence it is advisable to perform computing near the datasets. Hence, HDFS supports the feature of process portability over moving the large datasets. This feature of migrating the process or computing closer to the huge data sets reduces the network congestion issues and improves the data migration errors, thereby improving the overall throughput of the system. The HDFS documentation is updated on the URL http://hadoop.apache.org/hdfs/.

NameNode and the DataNodes: Being distributed design, the HDFS has a Java supported client-server architecture. The master server is a single NameNode which manages the file system with necessary access permissions and authentication for the client accesses. There can be many DataNodes usually one per node in a cluster. Any hardware that supports Java can be used for creating NameNode and DataNodes. These DataNodes are important local storage managers that exciters at the node side. A namespace is created for the file system and user data is stored in the files. A file created on such a namespace may be split into one or more data blocks which are then stored in the DataNodes. The DataNodes are responsible for the read and write operations request from the clients on the data blocks allocated to it. The DataNodes are also responsible for creating, deleting and replicating the data block upon the instructions from the NameNode.

The NameNode does the mapping of these data blocks to the DataNodes in addition to the file handling operations like opening, closing and renaming the files and the directories. NameNode is the arbitrator and repository for all HDFS meta-data. Figure A1.2 shows the HDFS architecture.

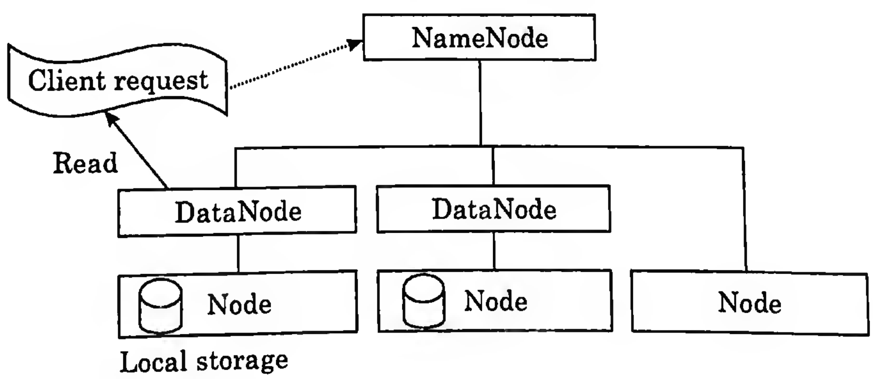


Figure A1.2 HDFS architecture.

Since GNU/Linux derivative operating systems are supported, traditional hierarchical file systems with namespaces are also supported by HDFS. HDFS does not support hard links and soft links to the datasets. The NameNode manages and maintains the namespaces in the cluster. The dataset or file replication is possible with meta-data description based on the replication requirement specified by the application. The data replication pipeline is maintained. Usually, the one-to-one node-DataNode mapping is maintained and managed by the NameNode. The data blocks on the physical storage/memory storage are equal in size except the last block.

Typical block size maintained by the HDFS is 64 MB, called as 64 MB data chunk, and data chunks may be organized on one or more DataNodes.

The data replication pipeline functions with configuring the data replication size, say 3. Then the data received does not reach directly to the NameNode but it fills the block on a node, the data is received using data packets of size 4 KB and then pipes it to another node selected by the NameNode for the replication and this process continues till the replication factor is reached. Then next block is taken for the processing and this continues till all the data is transferred. At the completion of the data storage and required replication, the metadata maintained by the NameNode is refreshed.

The cluster re-balancing is done by the HDFS when a free space of a DataNode falls below the threshold configured by reorganization of the data to a relatively free DataNode. Any client request or re-balancing verifies the check-sum for data validation. FSImage and the EditLog are two data structures maintained by the HDFS for metadata authentication. These data structures are replicated by the NameNode to avoid failure in the metadata which in case of failure use other copies to retrieve the session and keep the system healthy.

The FS shell is the command line interface provided by the HDFS to allow user to transact with the data. Table A1.1 shows some of the FS shell commands.

Table A1.1 FS Shell Command Illustration

FS Command Illustration	*Description*
bin/hadoop dfs-mkdir/MyWorkDir	Create a directory named/MyWorkDir
bin/hadoop dfs-rmr/MyWorkDir	Remove a directory named/MyWorkDir
bin/hadoop dfs-cat/MyWorkDir/myfile.txt	View the contents of a file named/MyWorkDir/myfile.txt
FS Admin Commands	
bin/hadoop dfsadmin-safemode enter	Put the cluster in Safe-mode
bin/hadoop dfsadmin-report	Generate a list of DataNodes
bin/hadoop dfsadmin-refreshNodes	Refresh or Update DataNode(s)

MapReduce Framework

With the introduction of HDFS and its distributed features, it is understood that the computing with datasets is a challenging task on Hadoop system. MapReduce is a software framework using which writing applications for fault tolerant computational processing on huge datasets of multi-terabytes or Big Data processing with dataset in peta-bytes in parallel and in large clusters of multiple thousand nodes is possible on a commodity hardware.

The dataset is subdivided to form the computable chunks. These chunks are independent in nature. The computation happens in a concurrent manner. The outcomes generated are sorted which are the given input to the reduced tasks. The files are used for Input/Output purpose. The job queues are maintained which are currently under execution, under sleep state or failed state and such tasks are scheduled for the execution. For small clusters, typically, has the computing node and the storage nodes shared on the same node. In other words, HDFS and the MapReduce applications are located and executed on the same node. This makes bandwidth space available.

The inputs are processed by the MapReduce framework applications and outputs are stored in the format <key, value> and supplied as the input to reduced processes using <key, value> format. The input–output overlapping windows help to reduce dataset transfer and make the bandwidth available. Since these tables are very large, the key and value classes must be serializing the dataset resulting in writable interfaces.

To illustrate it further, let us take one simple example where 10 numbers of value 2 are added; this may take number of iterations but 10 maps of '+' with 2 are reduced to value 10 and reduce to map it to one * resulting in 10 * 2 as one operation. The mapper and reducer process work with <key, value> pair to create the intermediate <key, value>, in other words, the input records are transferred to intermediate records. The MapReduce framework spawns one map task per input subdivide for multiple jobs using Job.setMapperClass (Class) method. Then, the map (WritableComparable, Writable, Context) is called per divide. The cleanup (Context) is called for any cleanup requirements. These intermediate maps are called with context write (WritableComparable, Writable). The iterations are counted for statistical purpose.

The reducer calls the reduce (WritableComparable, Iterable<Writable>, Context) method for each sub-divided <key, (list of values)> pair in the grouped inputs context. These unsorted outputs generated are written to the FileSystem using Context.write (WritableComparable, Writable). If the tasks are further not reducible, the number of reduced tasks is set to zero.

In a distributed and concurrent processing, memory always plays very important role. The virtual memory is required for the computing which can be specified in MegaBytes (MB) by the users or admins of MapReduce using mapreduce.{map|reduce}.memory.mb per process limit. The value assigned must not be less than the limit specified by -Xmx and is passed to JavaVM, else VM might not start.

The MapReduce framework has two major components; viz. single master ResourceManager, one slave as NodeManager and per application one MRAppMaster. These units collectively work synchronously using number of data structures for successful operations.

The latest revision of MapReduce is referred as MapReduce2.0 or MRv2 or Apache YARN. The major characteristic of YARN as compared to the conventional MapReduce is to split the Job Tracker functioning into two daemons for resource management and job scheduling respectively. The ResourceManager (RM) and the NodeManager (NM) form the computation network system. The ResourceManager controls the application and the system. Before submitting the Job for execution, the ApplicationMaster (AM), a framework specific library, demands the resources by negotiating resources from the ResourceManager and associates with the NodeManager(s) to execute and monitor the tasks. The ResourceManager has two main modules: Scheduler and ApplicationsManager. The Scheduler plays major role in allocating resources of the various running applications by computing the parameters such as capacities and queues. The ApplicationsManager performs functions of accepting job-submissions, creating the first container for executing the application specific ApplicationMaster and provides the service for restarting the ApplicationMaster container in case of failure.

◇ APACHE HADOOP ECO-SYSTEM APPLICATIONS

Pig

The Apache Pig, a platform for analysis of very large datasets, consists of high level language

for data analysis programmes. The Pig has a compiler that produces sequences of MapReduce programs, for which large-scale parallel implementations already exist. Pig's language consists of a textual language, called Pig Latin, which has the key properties such as Ease of programming, Optimization and Extensibility. The Pig can be used with local mode and MapReduce mode.

Hive

The Apache Hive is software that supports data warehouse applications, facilitating, querying and managing large datasets stored in distributed storage. Hive queries use SQL-like language, called HiveQL. This language also allows traditional Map/Reduce programmers to plug in their custom mappers and reducers when it is inconvenient or inefficient to express this logic in HiveQL.

SQOOP

Apache Sqoop is software designed for efficiently transferring bulk data between Hadoop and structured relational databases.

◇ Introduction to Flumes

Apache's Flume is an application projected by Apache for distributed and reliable system for efficiently collecting, aggregating and moving large amounts of log data from distributed sources to a centralized data store. A Flume event is called when the log data movement occurs. This event results in data flow having a byte payload and set of string attributes. These attributes are optional. This event is a JVM process that hosts the data components through which the events flow from external source to the next hop or destination.

◇ Introduction to Zookeeper

Apache's ZooKeeper is an open source server project that provides synchronization between the centralized infrastructure and the services to a Hadoop clusters by holding the common objects needed for the large cluster environments. These common objects hold information such as configuration information, hierarchical naming space including name services, group services, synchronization services and other application driven common objects required for the Hadoop cluster to function.

The ZooKeeper server maintains the status state of the system information using log files and in-memory per process regions. Very large Hadoop clusters are maintained by multiple Zookeeper servers using hierarchical structure, and the client machines can communicate data or information status to any one of the ZooKeeper server to retrieve or update the synchronization information. The application can create a file that persists in memory of a Zookeeper server, called znode. These znodes are watched by the servers to synchronized information related to the application. Any node in the cluster can update the znode. The changes so updated are communicated to all the registered nodes in the cluster. Any node in the cluster can register

to the znode for such updates. Hence, using these znodes the applications under the Hadoop framework maintain the synchronization of their tasks across the distributed cluster. Such updates go to the upper levels of the hierarchical structure with the help of cluster-wide status centralization service for managing and serializing the tasks across the servers in the distributed environment.

The latest version of the ZooKeeper can be installed from the Apache's website https://zookeeper.apache.org/. The ZooKeeper server can be started from the ZooKeeper directory path by running the script command:

/bin/zkServer.sh start

This is followed by evoking the CLI Manager from one of the ZooKeeper Server machine using the command:

/bin/zkCli.sh server
zkserver1.abc123.com:2181, zkserver2.abc123.com:2181, zkserver3.abc123.com:2181

This list of server of abc123.com on the port 2181 are supplied by the CLI Manager, and one of the servers is chosen for the connection. In case the operational connection is lost, then the supplied list is selected as a server for further communication and the clients sessions are transferred to this newly assigned server. This information is preserved in *zoo.cfg* file and it is very important that the port used must be open in all the machines supplied by the CLI Manager. If the client's session is initiated, then the creation of znode with edit and delete features is possible. The znode can be created by the command *create/my_znode*. The 'Hello World!' message my_znode gets broadcasted on the 127.0.0.1:2181 is locally hosted connection. The *rmr/my_znode* instruction removes the my_znode.

There are two major features of ZooKeeper. The first one is that the ZooKeeper is ordered. Each update is stamped by the ZooKeeper with a number identifier that reflects the order of all ZooKeeper transactions, supporting the synchronization resulting in sequential consistency. Another feature of ZooKeeper is fast workloads. It is more efficient in 'read-dominant' type workloads with average 10 time improvements in read-loads. This results in reliable, and timely performed processing.

The ZooKeeper's name space is much like the standard open source file system starting with (/). In the conventional file system, the leaf nodes are data nodes, whereas the ZooKeeper's node may hold data along with the path link. The data may hold the portable and tiny information like status information, configurations and location information. The hierarchical Access Control List (ACL), time-stamped updates and changes are maintained by the znode.

Table A1.2 presents the ZooKeeper command summary.

Table A1.2 ZooKeeper Command Summary

Command	*Description*
create	Creates a node at a location in the tree
delete	Deletes a node from the tree
exists	Tests if a node exists at a location of the tree

Command	*Description*
get data	Reads the data from a node
set data	Writes data to a node
get children	Retrieves a list of children of a node
Sync	Waits for data to be propagated

Applications of ZooKeeper

Zookeeper being a distributed system, has very large set of versatile applications. The Hadoop exploits the ZooKeeper when HDFS name-node fail-out condition and ensuring high availability of the YARN resource manager. The Hbase, a distributed database used in Hadoop uses it for the master selection, selection of region servers and its communication. The Neo4j is a distributed graph database uses ZooKeeper for master selection and read slave coordinates. Other Apache applications such as Solr Mesos also use ZooKeeper.

◇ BIG DATA MINING WITH HADOOP

The business success, nowadays, mainly depends on the ability to store and analyze the large datasets or Big Data. The analytical intelligence results are expected out of Big Data, the raw data or datasets are processed with the intention of mining. Hence, once-write-once-read-many type of features of Hadoop are very useful in Big Data mining. The distributed and parallel processing of dataset on conventional machines is another big advantage of the Hadoop for Big Data mining. Another advantage of process portability rather than the dataset portability feature is very useful for Big Data mining. Since, multiple petabytes of dataset can be processed by migrating the computing rather than the dataset, it can save lot of network bandwidth and can generate time efficient responses. Since, different intention can be applied simultaneously on the Big Datasets, the business intelligence has numerous applications of Big Data processing with Hadoop framework.

Annexure II

Installing and Running GATE

—Dr. Yashodhara Haribhakta

◇ 1. PREREQUISITES NEEDED FOR GATE

Java 2 environment should be installed beforehand.

(i) GATE 3.1 with version 1.4.2.
(ii) GATE 4.0 beta 1 or later with version 5.0.
(iii) GATE 6.1 or later with version 6.0.

◇ Installation

Most stable and running version of GATE can be found at http://gate.ac.uk/download/.

◇ 2. HOW TO RUN GATE?

◇ For Linux Users

(i) Download the GATE tool from http://gate.ac.uk/download/.
(ii) Extract the Zip folder.
(iii) Go to bin and run the gate.sh file.
(iv) To run using command line, run the command./gate.sh on the terminal.

◇ For Windows Users

(i) Download the GATE tool from http://gate.ac.uk/download/.
(ii) Extract the Zip folder.
(iii) Run the gate.exe file.

◇ 3. FEATURES OF GATE

(i) GATE includes ANNIE (A nearly New Information Extraction System).
(ii) Various plugins are available in GATE for machine learning, quering, POS tagging, etc.
(iii) Annotations on text are manipulated by JAPE transducer.
(iv) GATE processes documents of various formats PDF, ODT, HTML, etc.

◇ 4. IMPORTANT TERMS AND DEFINITION

(i) **Corpus:** It is a set of files bundled together for running GATE plugins over it. Document class in Java is member of Corpus.
(ii) **Annotations:** Annotations that are created on documents, e.g., annotation for Organization.

Entity [Annotation Impl: id: ID given to the annotation by IDE; type = Type of the Annotation like Person, Organization, Date, Time, etc.; features = Rules from Named Entity JAPE rules which have matched with the given annotation; start node offset, end node offset]
(iii) **Annotation sets:** It comprises groups of annotations.
(iv) **Applications:** Groups of processes to be run on a document or corpus.
(v) **DataStores:** Saved processed documents and resources.
(vi) **Processing resources (PR):** It is used for manipulating documents with respect to annotations and consists of number processing resources arranged in a sequence.
(vii) **Language resources (LR):** Corpus and documents are of Language Resource type and it has a FeatureMap (Java class) associated with it which holds attribute and value information of the resource.

◇ 5. RUNNING GATE IDE

Run the gate.sh file for Linux and gate.exe file for Windows. A main window of GATE IDE will appear as shown in Figure A2.1.

◇ 6. HOW TO CREATE A LANGUAGE RESOURCE?

(i) Right click on language resources.
(ii) New → GATE document.
(iii) Select the required file or you can type a string there.
(iv) Give a name to your document if you want.
(v) Click OK.

Figure A2.2 shows creation of Language resource.

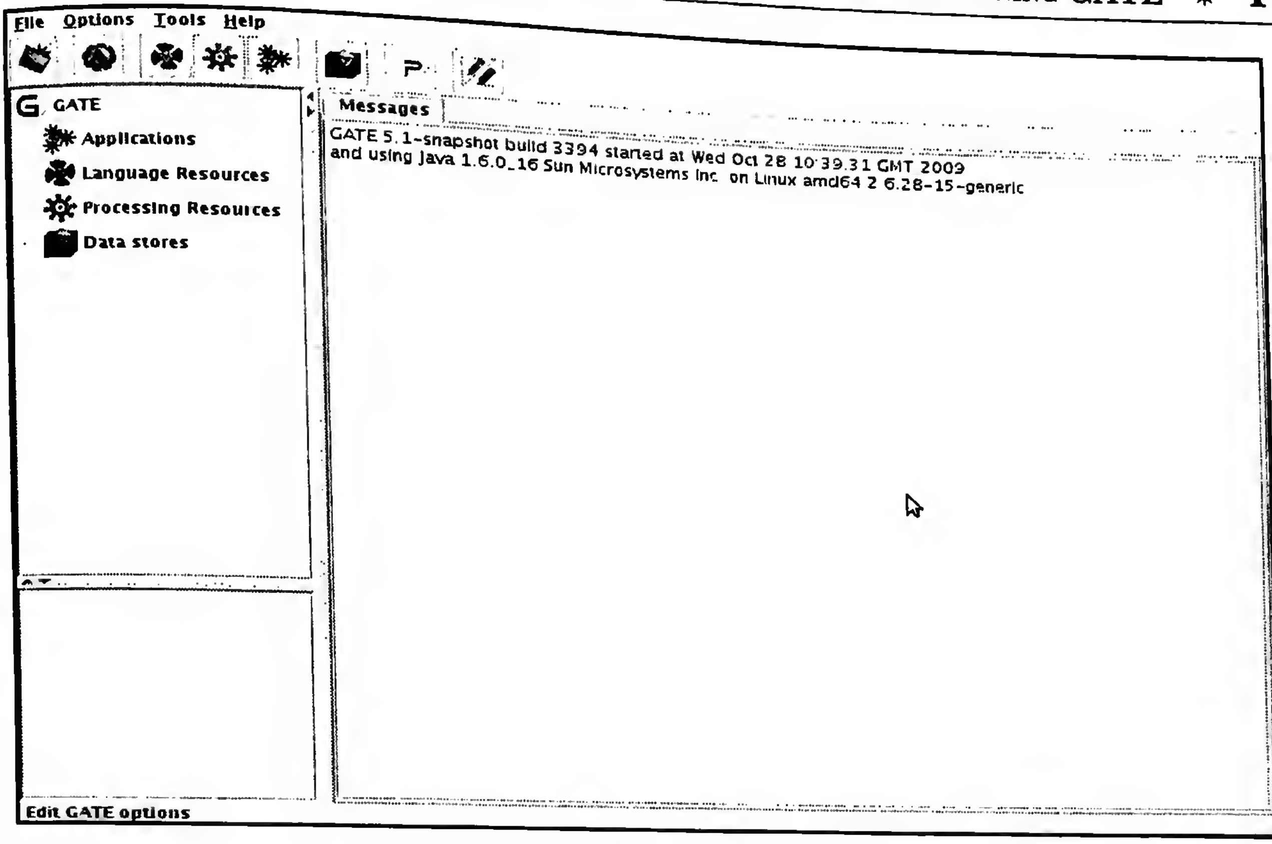


Figure A2.1 Main GATE window.

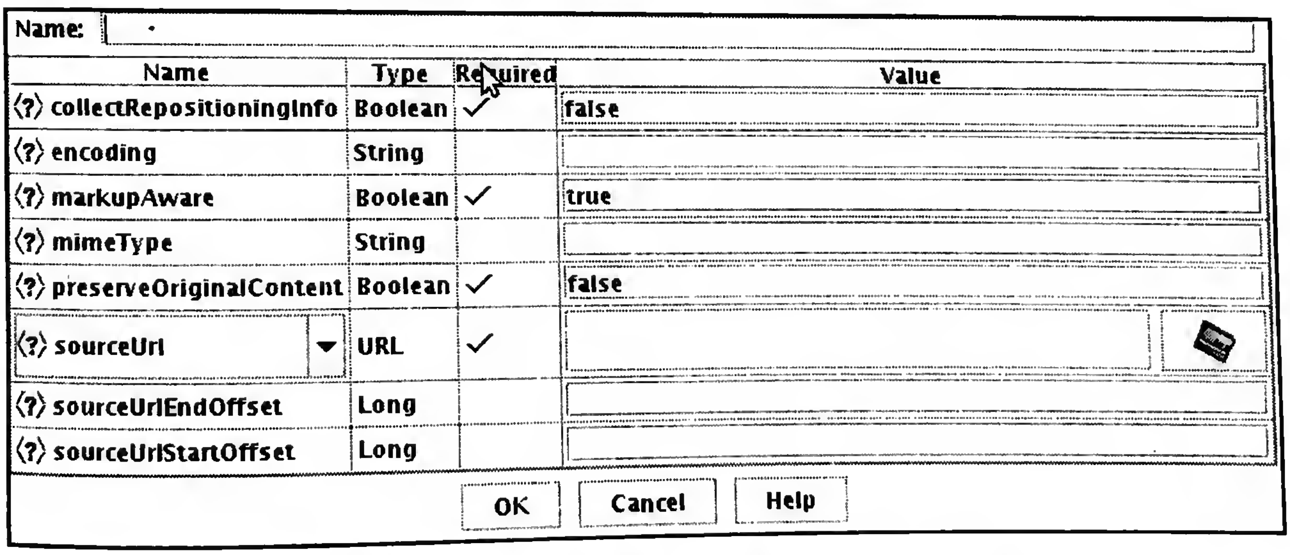


Figure A2.2 Creating a language resource.

◇ 7. HOW TO CREATE A CORPUS?

Method 1

(i) Right click on Language resources.
(ii) New → GATE corpus.
(iii) Add required documents.
(iv) Click OK.

Method 2

(i) Right click on document under LR.

(ii) Select New corpus with this document as shown in Figure A2.3.

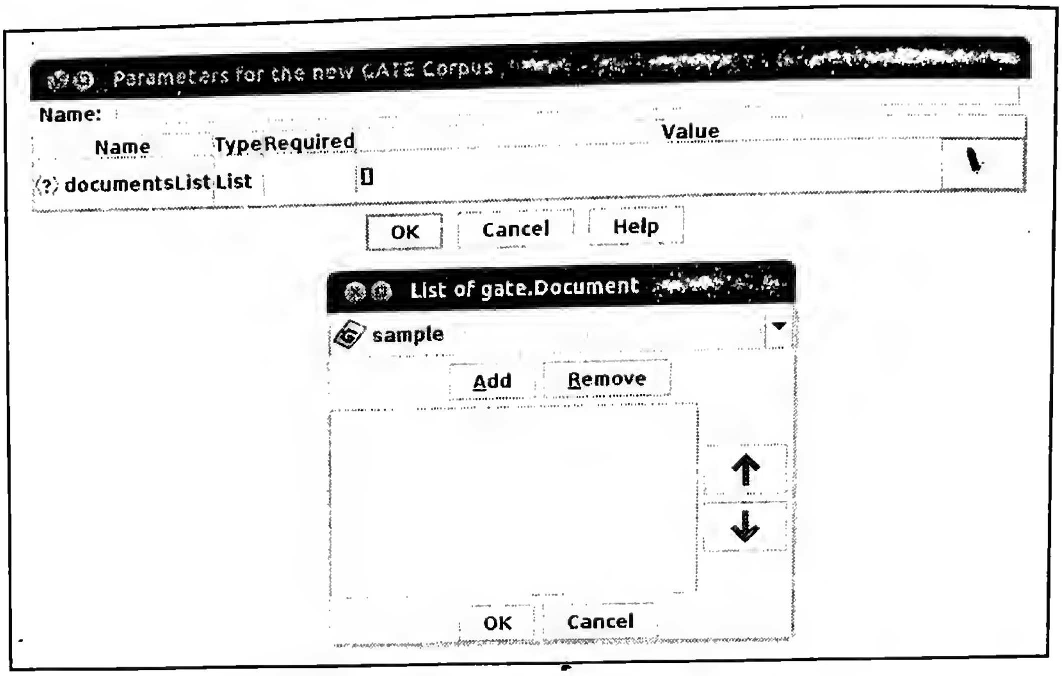


Figure A2.3 Creating a corpus.

◇ 8. HOW TO ADD NEW PLUGINS?

(i) Go to file.

(ii) Go to manage Creole Plugins. Figure A2.4 shows the Creole Plugins window.

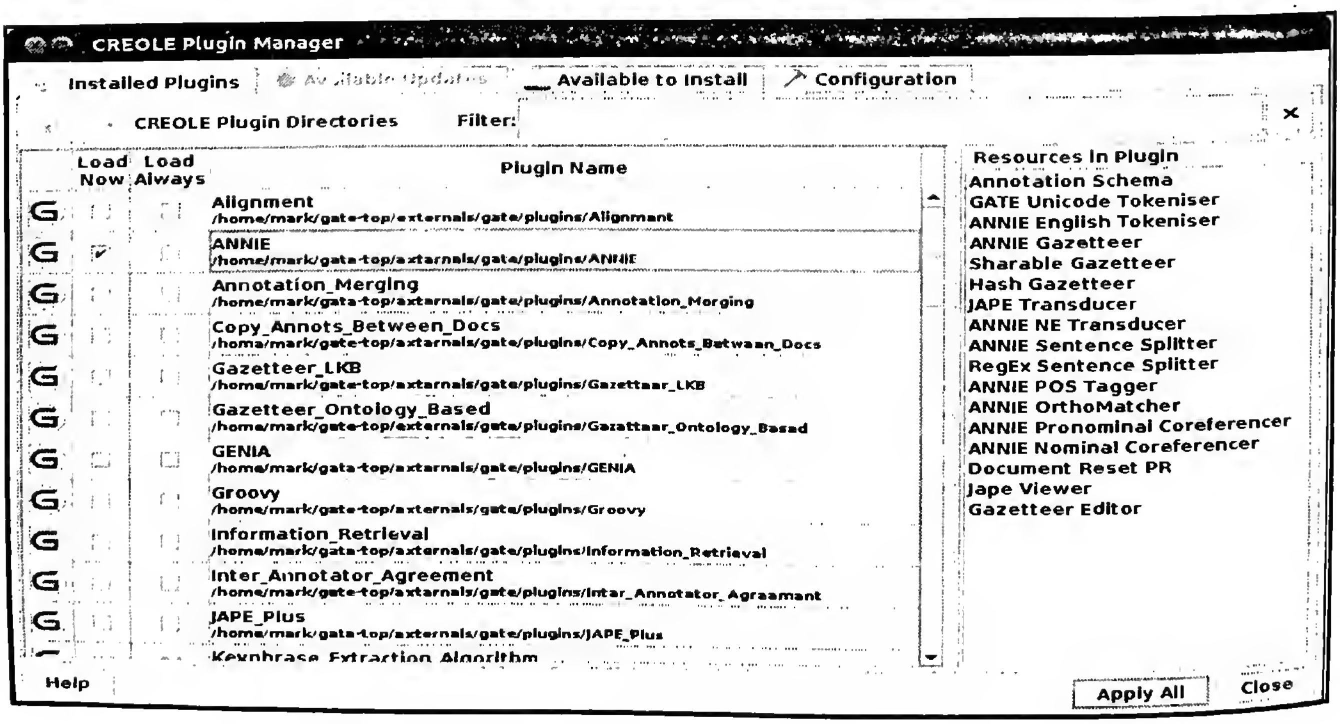


Figure A2.4 Creole plugins window.

Bibliography

Aggarwal, Charu C. and ChengXiang Zhai, "Mining Text Data".

Agrawal, R. and Srikant, R., "Fast Algorithms for Mining Association Rules," Proc. 20th Int. Conf. Very Large Data Bases, *VLDB*, Vol. 1215, pp. 487–499, 1994.

Agrawal, R., et al., "Automatic Subspace Clustering of High Dimensional Data for Data Mining Applications," *Proceedings of the SIGMOD*, Vol. 27, Issue 2, pp. 94–105, 1998.

Anick, P., and Vaithyanathan, S., "Exploiting Clustering and Phrases for Context Based Information Retrieval," *ACM SIGIR Conference*, 1997.

Attardi, G., DiMarco, S., and Salvi, D., "Categorization by Context," *Journal of Universal Computer Science*, Vol. 4, No. 9, pp. 719–736, 1998.

Beil, F., Ester M., and Xu, X., "Frequent Term-based Text Clustering," *Proceedings of the Eighth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pp. 436–442, 2002.

Berchtold, S., et al., "A Cost Model for Nearest Neighbour Search in High Dimensional Data Space," Proceedings of the 16th Symposium on Principles of Database Systems (PODS), pp. 78–86, 1997.

Beyer, K., et al., "When is Nearest Neighbors Meaningful?" Proceedings of 7th International Conference on Database Theory (ICDT-1999), Jerusalem, Israel, pp. 217–235.

Bhatotia, Pramod, et al., "Incoop: MapReduce for Incremental Computations," *Proceedings of the 2nd ACM Symposium on Cloud Computing*, ACM, 2011.

Blackman, Josh, Sokol, L., and Chan, S., "Context-Based Analytics in a Big Data World: Better Decisions," A Report of IBM. http://joshblackman.com/blog/2013/08/07/how-does-facebook-decide-what-to-show-you/, 2013.

Blanco, Rio and Lioma, Christina, "Graph-based Term Weighting for Information Retrieval," *Information Retrieval*, Vol. 15, No. 1, pp 54–92, February 2012.

Blei, D.M., Ng, A.Y., and Jordan, M.I., "Latent Dirichlet Allocation," *Journal of Machine Learning Research*, Vol. 3, pp. 993–1022, 2003.

Blum, A. and Langley, P., "Selection of Relevant Features and Examples in Machine Learning," *Artificial Intelligence*, Vol. 97, pp. 245–271, 1997.

Brezillon, Patrick, "Context in Problem Solving: A Survey," *The Knowledge Engineering Review*, Cambridge University Press, Vol. 14, No. 1, pp. 47–80, 1999.

Brown, Peter J. and Bovey, John D. and Chen, Xian, "Context-aware Applications: From the Laboratory to the Marketplace," *IEEE Personal Communications*, Vol. 4, No. 5, pp. 58–64, 1997.

Buhl, H.U., et al., "Big Data," *Business and Information Systems Engineering*, Vol. 5, No. 2, pp. 65–69, 2013.

Byron, Spice, "CMU Research Finds Regional Dialects Are Alive and Well on Twitter," http://www.cmu.edu/news/archive/2011/January/jan7_twitterdialects.shtml, 2011.

Byron, Tau, Obama Campaign Final Fundraising total: $1.1 billion, Politico, January 19, 2013, http://www.politico.com/story/2013/01/obama-campaign-final-fundraising-total-1-billion-86445.html

Charniak, E., "A Maximum-entropy-inspired Parser," Proceedings of the 1st North American Chapter of the Association for Computational Linguistics Conference, pp. 132–139, 2000.

Chen, Guanling, et al., "A Survey of Context-aware Mobile Computing Research," Technical Report TR2000-381, Department of Computer Science, Dartmouth College, 2000.

Chen, M., Mao, S., and Liu, Y., "Big Data: A Survey," *Mobile Networks and Applications*, Vol. 19, No. 2, pp. 171–209, 2014.

Church, K. and Gale W., "Poisson Mixtures," *Nat. Lang. Eng.*, Vol. 1, No. 2, pp. 163–190, 2004.

Church, K. and Mercer, R., "Introduction to the Special Issue on Computational Linguistics using Large Corpora," *Computational Linguistics*, Vol. 19, No. 1, pp. 1–24, 1993.

Church, K. and Thiesson, B., "The Wild Thing!," *Proceedings of the ACL*, pp. 93–96, 2005.

Church, K.W., "A Stochastic Parts Program and Noun Phrase Parser for Unrestricted Text," *Proceedings of the Second Conference on Applied Natural Language Processing*, pp. 136–143, 1998.

Cutting, D., et al., "Scatter/Gather: A Cluster-Based Approach to Browsing Large Document Collections," Proceedings of the 15th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, June 21–24, Copenhagen, Denmark, pp. 318–329, 1992.

Cutting, D., et al., Scatter/Gather: A Cluster-based Approach to Browsing Large Document Collections, ACM SIGIR Conference, 1992.

Dao, N.D., "A New Class of Functions for Describing Logical Structures in Text," Doctoral Dissertation, Massachusetts Institute of Technology, 2004.

Daud, A., et al., "Knowledge Discovery through Directed Probabilistic Topic Models: A Survey," 2008.

David, Ogilvy, We Sell or Else, Ogilvy and Mather, https://www.youtube.com/watch?v=Br2KSsaTzUc

Daxin, J., Tang, C., and Zhang, A., "Cluster Analysis for Gene Expression Data: A Survey," *IEEE Transaction on Knowledge and Data Engineering*, Vol. 16, Issue 11, pp. 1370–1386, 2004.

Dey, A., Abowd, G., and Salber, D., "A Conceptual Framework and a Toolkit for Supporting the Rapid Prototyping of Context-aware Applications," *Human–Computer Interaction*, Vol. 16, No. 2–4, pp. 97–166, 2001.

Dey, A.K., "Understanding and Using Context," *Personal and Ubiquitous Computing*, Springer-Verilag, Vol. 5, No. 1, pp. 4–7, 2001.

Diallo, B.A.A., et al., "Mobile and Context-aware GeoBI Applications: A Multilevel Model for Structuring and Sharing of Contextual Information," *Journal of Geographic Information System*, Vol. 4, No. 5, 425, 2012.

Dolan, Yonatan and Razon, Oren, "Using Apache Hadoop for Context-aware Recommender Systems," IT@Intel White Paper, February 2014.

Doug, Laney, Deja VVVu: Others Claiming Gartner's Construct for Big Data, Gartner blog, January 14, 2012, http://blogs.gartner.com/doug-laney/deja-vvvue-others-claiming-gartners-volume-velocity-variety-construct-for-big-data/

Douglas, K., "Infographic: Big Data Brings Marketing Big Numbers," 2012, http://www.marketingtechblog.com/ibm-big-datamarketing/.

Dredge, Stuart, "How does Facebook Decide What to Show in my News Feed?" http://www.theguardian.com/technology/2014/jun/30/facebook-news-feed-filters-emotion-study, 2014.

Dumitrescu, Alexandra and Santini, Simone, "Think Locally, Search Globally: Context-based Information Retrieval," *IEEE International Conference on Semantic Computing*, pp. 396–401, 2009.

Erkan, Gunes and Radev, Dragomir, R., "LexRank: Graph-based Centrality as Salience in Text Summarization, *Journal of Artificial Intelligence Research*, Vol. 22, Issue 1, pp. 457–479, 2004.

Frank, I.E. and Todeschini, R., *The Data Analysis Handbook*, Elsevier Science, 1994.

Friedman, J., "An Overview of Computational Learning and Function Approximation," In: *From Statistics to Neural Networks: Theory and Pattern Recognition Applications* (Cherkassky, Friedman, Wechsler, Eds.) Springer-Verlag 1, 1994.

Fukunaga, K., *Introduction to Statistical Pattern Recognition*, Academic Press, New York, 1990.

Gao, J., Kwan, P.W. and Guo, Y., "Robust Multivariate L1 Principal Component Analysis and Dimensionality Reduction," *Elsevier's Neurocomputing*, Vol. 72, Issue 4–6, pp. 1242–1249, 2009.

Gawrysiak, P., Gancarz, L., and Okoniewski, M., "Recording Word Position Information for Improved Document Categorization," *Proceedings of the PAKDD Text Mining Workshop*, 2002.

Genc, Yegin, et al., "Discovering Context: Classifying Tweets through a Semantic Transform-based on Wikipedia," Springer's Foundations of Augmented Cognition, Directing the Future of Adaptive Systems, pp. 484–492, 2011.

Gildea, D. and Jurafsky D., "Automatic Labeling of Semantic Roles," *Comput. Linguist.*, Vol. 28, No. 3, pp. 245–288, 2002.

Grimes, S., "Unstructured Data and the 80 Percent Rule," Clarabridge Bridgepoints, 2008.

Guha, S., Rastogi, R., and Shim, K., CURE: An Efficient Clustering Algorithm for Large Databases, *ACM SIGMOD Conference*, 1998.

Hadoop, A., "Hadoop," 2009, http://hadoop.apache.org/.

Hammer, Barbara, He, Haibo, and Martinetz, Thomas, *Learning and Modelling Big Data*, ESANN 2014 Proceedings, European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning, Bruges (Belgium), pp. 23–25, April 2014.

Han, J. and Kamber, M., *Data Mining: Concepts and Techniques*, Morgan Kaufmann, 2000.

Haveliwala, T.H., "Topic Sensitive Pagerank," *ACM Comput. Surv.*, Vol. 34, pp. 1–47, March 2002.

Hearst, Marti A., "Multi-paragraph Segmentation of Expository Text," Proceedings of the 32nd Annual Meeting on Association for Computational Linguistics, pp. 9–16, 1994.

Hofmann, T., "Probabilistic Latent Semantic Indexing," Proceedings of the 22nd Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, *SIGIR '99*, ACM, pp. 50–57, 1999.

Holzinger, C. Stocker, et al., "Combining HCI, Natural Language Processing, and Knowledge Discovery—Potential of IBM Content Analytics as an Assistive Technology in the Biomedical Field," In: *Human–Computer Interaction and Knowledge Discovery in Complex, Unstructured, Big Data*, Vol. 7947 of *Lecture Notes in Computer Science*, pp. 13–24, Springer, Berlin, Germany, 2013.

Hull, Richard, Neaves, Philip, and Bedford-Roberts, James, "Towards Situated Computing," *IEEE First International Symposium on Wearable Computers,* Digest of Papers, pp. 146–153, 1997, http://pearanalytics.com/blog/2009/twitter-study-reveals-interesting-results-40-percent-pointless-babble/

Jain, A. and Dubes, R., *Algorithms for Clustering Data*, Prentice Hall, 1988.

Jang, C., et al., "Text Classification using Graph Mining-based Feature Extraction," *Research and Development in Intelligent Systems XXVI*, Springer, pp. 21–34, 2010.

Jolliffe, I.T., *Principal Component Analysis*, Springer, October 2002.

Joshi, Prachi and Kulkarni, Parag, "Incremental Learning: Methods and Techniques—A Survey," *IJDKP*, 2012.

Joshua, Green, The Science Behind Those Obama Campaign E-Mails, *Bloomberg BusinessWeek*, November 29, 2012, http://www.businessweek.com/articles/2012-11-29/the-science-behind-those-obama-campaign-e-mails

Kaufman, L. and Rousseeuw, P.J., *Finding Groups in Data: An Introduction to Cluster Analysis*, John Wiley & Sons, New York.

Khan, Nawsher, et al., "Big Data: Survey, Technologies, Opportunities, and Challenges," *Scientific World Journal*, Vol. 2014, Article 712826, p. 18.

Klapaftis, I.P. and Manandhar, S., "Unsupervised Word Sense Disambiguation using the www," Proceedings of the 2006 Conference on STAIRS 2006: Proceedings of the Third Starting AI Researchers' Symposium, IOS Press, pp. 174–183, 2006.

Kleinberg, "Bursty and Hierarchical Structure in Streams," *Proceedings of KDD '02*, pp. 91–101, 2002.

Ko, Y., Park, J., and Seo, J., "Improving Text Categorization Using the Importance of Sentences," *Information Processing and Management*, Vol. 40, No. 1, pp. 65–79, 2004.

Kriegel, H.P., Kroger, P. and Zimek, A., "Clustering High-Dimensional Data: A Survey on Subspace Clustering, Pattern-Based Clustering, and Correlation Clustering," *ACM Transactions on Knowledge Discovery from Data (TKDD)*, Vol. 3, Issue 1, Article 1, 2009.

Kulkarni, Anagha, Tokekar, Vrinda, and Kulkarni, Parag, "Discovering Context of Labeled Text Documents using Context Similarity Coefficient," *Procedia Computer Science 49C*, 2015.

Kulkarni, Anagha, Tokekar, Vrinda, and Kulkarni, Parag, "Discovering Context using Contextual Positional Regions based on Chains of Frequent Terms in Text Documents," *Intelligent Systems Technologies and Applications*, Springer International Publishing, pp. 321–332, 2016.

Kulkarni, Anagha, Tokekar, Vrinda, and Kulkarni, Parag, "Text Classification by Enhancing Weights of Terms-based on their Positional Appearances," *International Journal of Computer Applications*, Vol. 78, No. 9, pp. 23–26, 2013.

Kulkarni, Parag, *Reinforcement and Systemic Machine Learning for Decision Making*, John Wiley & Sons, 2012.

Lars, Mieritz, Gartner Survey Shows Why Projects Fail, this is what good looks like, June 1, 2012, http://thisiswhatgoodlookslike.com/2012/06/10/gartner-survey-shows-why-projects-fail/

Lee, M. and Park, C.H., "On Applying Dimensionality Reduction for Multi-labeled Problems," In: *Lecture Notes of International Conference MLDM*, LNAI 4571, pp. 131–143, 2007.

Li, C., Sun, A., and Datta, A., "Twevent: Segment-based Event Detection from Tweets," *Proceedings of the 21st ACM International Conference on Information and Knowledge Management*, pp. 155–164, 2012.

Li, M.L.Z., Wang, B. and Ma, W.-Y., "A Probabilistic Model for Retrospective News Event Detection," *Proceedings of SIGIR '05*, pp. 106–113, 2005.

Lin, D. and Pantel, P., "Concept Discovery from Text," 2002 Gate tool: https://gate.ac.uk/. Jape found at user guide of GATE tool.

Lin, D. and Pantel, P., "Concept Discovery from Text," Proceedings of the 19th International Conference on Computational Linguistics, pp. 1–7, 2002.

Lin, D., "Automatic Retrieval and Clustering of Similar Words," in Proceedings of the 17th International Conference on Computational Linguistics, Vol. 2, ACL, pp. 768–774, 1998.

Liu, H. and Motoda, H., *Feature Selection for Knowledge Discovery and Data Mining*, Kluwer Academic Publishers, Boston, 1998.

Liu, Yunhuai, et al., "Semantic Link Network Based Model for Organizing Multimedia Big Data." *IEEE Transactions on Emerging Topics in Computing*, Vol. 2, No. 3, pp. 376–387, 2014.

Malik, H.H. and Kender J.R., "Classification by Pattern-based Hierarchical Clustering," In: *From Local Patterns to Global Models Workshop*, ECML/PKDD, 2008.

Manning, C.D., Raghavan, P., and Schütze, H., *Introduction to Information Retrieval*, Cambridge: Cambridge University Press, Vol. 1, 2008.

Manyika, J., et al., "Big Data: The Next Frontier for Innovation, Competition, and Productivity," *Tech. Rep.*, McKinsey, May 2011.

Mei, Q. and Church, K., "Entropy from Search Logs: How Hard is Search with Personalization with Backup," *Proceeding of WSDM '08*, pp. 45–54, 2008.

Mei, Q., Ling, X., and Zhai, C., "Topic Sentiment Mixture: Modeling Facets and Opinions in Weblogs," *Proceedings of WWW '07*, 2007.

Mihalcea, Rada and Tarau, Paul, TextRank: Bringing Order into Texts, Association for Computational Linguistics, EMNLP-04, pp. 404–411.

Miller, G.A., et al., "Introduction to WordNet: An Online Lexical Database," Vol. 3, No. 4, pp. 235–244, 1990.

Morris, Betsy, Steve Jobs Speaks Out, Fortune, March 7, 2008 http://archive.fortune.com/galleries/2008/fortune/0803/gallery.jobsqna.fortune/3.html

Murata, M., et al., "Japanese Probabilistic Information Retrieval using Location and Category Information," Proceedings of the Fifth International Workshop on Information Retrieval with Asian Languages, ACM, pp. 81–88, 2000.

Navrat, Pavol and Taraba, Tomas, "Context Search," IEEE/WIC/ACM International Conferences on Web Intelligence and Intelligent Agent Technology Workshops, pp. 99–102, 2007.

Ng, R. and Han, J., Efficient and Effective Clustering Methods for Spatial Data Mining, *VLDB Conference*, 1994.

Pantel, P. and Lin, D., "Automatically Discovering Word Senses," Proceedings of the 2003 Conference of the North American Chapter of the Association for Computational Linguistics on Human Language Technology: Demonstrations, Vol. 4, NAACL-Demonstrations '03, Association for Computational Linguistics, pp. 21–22, 2003.

Park, C.H. and Lee, M., "On Applying Linear Discriminant Analysis for Multi-labeled Problems," *Journal of Pattern Recognition Letters Archive*, Vol. 29, No. 7, pp. 878–887, 2008.

Parker, Charles, "Incremental Learning Algorithms for Fast Classification in Data Stream," IEEE, Machine Learning from Streaming Data: Two Problems, Two Solutions, Two Concerns, and Two Lessons, March 12, 2013.

Pasca, M., et al., "Names and Similarities on the Web: Fact Extraction in the Fast Lane. in acl-44," In: ACL-44: Proceedings of the 21st International Conference on Computational Linguistics and the 44th Annual Meeting of the Association for Computational Linguistics, pp. 809–816, 2006.

Pedersen, T. and Kolhatkar, V., "Wordnet::senserelate::allwords: A Broad Coverage Word Sense Tagger that Maximizes Semantic Relatedness," NAACL-Demonstrations '09, Association for Computational Linguistics, pp. 17–20, 2009.

Pena, J.M., et al., "Dimensionality Reduction in Unsupervised Learning of Conditional Gaussian Networks," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 23, Issue 6, pp. 590–603, 2001.

Pierrehumbert, H.J., "Teun A Van Dijk Text and Context: Explorations in the Semantics and Pragmatics of Discourse," *Journal of Linguistics*, Vol. 16, pp. 113–119, 1980.

Raez, M., PhD Thesis—Automatic Categorization of Documents in High Energy Physics Domain, Granada University, 2006.

Sahami, M., et al., AAAI-98 Workshop on Learning for Text Categorization, pp. 55–62, 1998.

Salton, G. and Buckley, C., "Term Weighting Approaches in Automatic Text Retrieval," *Information Processing and Management*, Vol. 24, No. 5, pp. 513–523, 1988.

Salton, G., *An Introduction to Modern Information Retrieval*, McGraw-Hill, 1983.

Schilit, B., Adams, N. and Want, R., "Context Aware Computing Applications," 1st International Workshop on Mobile Computing Systems and Applications, 1994.

Schilit, Bill N. and Thrimer, Marvin M., "Disseminating Active Map Information to Mobile Hosts," *IEEE Network*, Vol. 8, No. 5, pp. 22–32, 1994.

Sebastiani, F., "Machine Learning in Automated Text Categorization," *ACM Comput. Surv.*, Vol. 34, pp. 1–47, March 2002.

Seth, Grimes, Text Analytics 2014: User Perspectives on Solutions and Providers, Alta Plana Corporation, July 9, 2014, http://www.digitalreasoning.com/resources/Text-Analytics-2014-Digital-Reasoning.pdf

Singhal, A., Buckley, C., and Mitra, M., Pivoted Document Length Normalization, ACMSIGIR Conference, pp. 21–29, 1996.

Sokol, L. and Chan, S., "Context-Based Analytics in a Big Data World: Better Decisions," A Report of IBM, 2013.

Solur, Sridhar, *New Relic {Future} Talks*, November 2013.

Sonawane, S.S. and Kulkarni P.A., "Graph-based Representation and Analysis of Text Document: A Survey of Techniques," *International Journal of Computer Applications*, Vol. 96, No. 19, pp. 1–8, June 2014, published by Foundation of Computer Science, New York, USA.

Steyvers, M., Smyth, P., and Griffths, T., "Probabilistic Author-topic Models for Information Discovery," *Proceedings of KDD '04*, pp. 306–315, 2004.

Stovall, J.G., *Writing for Mass Media*, 6th ed., Pearson Education, 2006.

Strang, G., *Linear Algebra and its Applications*, 4th ed., Brooks/Cole Inida, 2005.

Subramaniam, L.V., "Big Data and Veracity Challenges," Text Mining Workshop, ISI Kolkata, January 2014.

Sun, Liang, et al., *Multi-Label Dimensionality Reduction*, Chapman and Hall, CRC Press, Taylor and Francis Group, 2013.

Tan, P.S., et al., "A Context Model for B2B Collaborations," IEEE International Conference on Services Computing, 2008.

Tatar, D., et al., "A Chain Dictionary Method for Word Sense Disambiguation and Applications," *CoRR*, Vol. abs/0806.2581, 2008.

Tsoumakas, G. and Katakis, I., "Multi-label Classification: An Overview," *International Journal of Data Warehousing and Mining*, Vol. 3, No. 3, pp. 1–13, 2007.

Valle, Kjetil and Ozturk, Pinar, Graph-based Representations for Text Classification, India Norway Workshop on Web Concepts and Technologies, 3 October 2011.

Vascellaro, Jessica E., "Turns Out Apple Conducts Market Research After All," *Wall Street Journal*, 26 July 2012, http://blogs.wsj.com/digits/2012/07/26/turns-out-apple-conducts-market-research-after-all/

Wan, K., "A Brief History of Context," arXiv preprint arXiv:0912.1838, 2009.

Wang, X. and Paliwal, "Feature Extraction and Dimensionality Reduction Algorithms and their Applications in Vowel Recognition," *Journal of Pattern Recognition*, Vol. 36, pp. 2429–2439, 2003.

Welling, M., Rosen-Zvi, M., and Hinton, G.E., "Exponential Family Harmoniums with an Application to Information Retrieval," In: *NIPS*, 2004.

Wigelius, H. and Vataja, H., "Dimensions of Context Affecting User Experience in Mobile Work," INTERACT'09: Proceedings of the 12th IFIPTC 13 International Conference on Human-Computer Interaction, Berlin, 2009.

Xinglin, L., et al., "Text Similarity Computing Based on Thematic Term Set," *International Journal of Advancements in Computing Technology*, Vol. 4, No. 6, 2012.

Xue, X.B. and Zhou, Z.H., "Distributional Features for Text Categorization," *Knowledge and Data Engineering, IEEE Transactions*, Vol. 21, No. 3, pp. 428–442, 2009.

Yan, Cairong, et al., "IncMR: Incremental Data Processing Based on Mapreduce." *Cloud Computing (CLOUD),* IEEE 5th International Conference, 2012.

Yap, Jamie, "Big Data Analysis Needs Human Context," http://www.zdnet.com/article/big-data-analysis-needs-human-context/, 2012.

Yu, L. and Liu, H., "Feature Selection for High Dimensional Data: A Fast Correlation Based Filter Solution," Proceedings of the Twentieth Int. Conf. on Machine Learning, pp. 856–863, 2003.

Zakor, J., "A Novel Context-based Technique for Web Information".

Zang, Wenyu, Zhang, Peng, Zhou, Chuan, and Guo, Li, "Comparative Study between Incremental and Ensemble Learning on Data Streams: Case study", *Journal of Big Data*, Springer, 2014.

Zhang, T., Ramakrishnan, R., and Livny, M., BIRCH: An Efficient Data Clustering Method for Very Large Databases, *ACM SIGMOD Conference,* 1996.

Index

BIG DATA ANALYTICS

Edited by Parag Kulkarni • Sarang Joshi • Meta S. Brown

The book is an unstructured data mining quest, which takes the reader through different features of unstructured data mining while unfolding the practical facets of Big Data. It emphasizes more on machine learning and mining methods required for processing and decision-making. The text begins with the introduction to the subject and explores the concept of data mining methods and models along with the applications. It then goes into detail on other aspects of Big Data analytics, such as clustering, incremental learning, multi-label association and knowledge representation. The readers are also made familiar with business analytics to create value. The book finally ends with a discussion on the areas where research can be explored.

The book is designed for the senior level undergraduate, and postgraduate students of computer science and engineering.

KEY FEATURES

- Contains numerous examples and case studies.
- Discusses Apache's Hadoop—a software framework that enables distributed processing of large datasets across the clusters of computing machines.
- Incorporates review questions, MCQs, laboratory assignments and critical thinking questions at the end of the chapters, wherever required.

ABOUT THE EDITORS

Parag Kulkarni, PhD (IIT Kharagpur), DSc (UGSM, Switzerland), is Founder and CEO of iknowlation Research Labs—a machine learning and data engineering product company. Dr. Kulkarni is a consultant in areas of machine learning and knowledge innovation to more than one dozen organizations. He has authored one dozen of books including *Knowledge Innovation Strategy, Artificial Intelligence* (Published by PHI Learning), and *Reinforcement and Systemic Machine Learning*, and more than 220 research papers. His areas of interest include artificial intelligence, machine learning, business and knowledge innovation and data mining.

Sarang Joshi, PhD, is Professor at Pune Institute of Computer Technology (PICT), Pune. He was the chairman of board of studies and member of academic council, Savitribai Phule Pune University (SPPU), formally Pune University. Professor Joshi's areas of interest include visualization, virtualization algorithms and data processing.

Meta S. Brown is President of A4A Brown Inc.—a boutique consultancy that helps technical people to communicate with executives and clients. She is the author of *Data Mining for Dummies*, and creator of the storytelling for Data Analysts and Tech Workshops. Her areas of interest include business analytics and text analytics.

(***Contributors***: Dr. Prachi Joshi, Dr. Sheetal Sonawane, Dr. Anagha Kulkarni, Dr. Yashodhara Haribhakta, Dr. Sonal Dharmadhikari, and Dr. Sunita Jahirabadkar)

ISBN:978-81-203-5116-5

9 788120 351165

₹ 250.00

www.phindia.com